

AgRISTARS

SM-G1-04086
TM 82121

NASA-TM-82121 19830019046

A Joint Program for
Agriculture and
Resources Inventory
Surveys Through
Aerospace
Remote Sensing

Soil Moisture

Technical Memorandum 82121

A COMPUTER PROGRAM FOR THE SIMULATION OF HEAT AND MOISTURE FLOW IN SOILS

P. Camillo and T.J. Schmugge

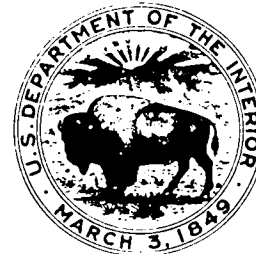
LIBRARY COPY

MAR 21 1983

LANGLEY RESEARCH CENTER
LINTHROPE, TEXAS
HAMPTON, VIRGINIA



NASA



GODDARD SPACE FLIGHT CENTER
Greenbelt, Maryland 20771

DISPLAY 16/2/1
83N27317*# ISSUE 16 PAGE 2591 CATEGORY 43 RPT#: E83-10342
NASA-TM-82121 SM-G1-04086 NAS 1.15:82121 CNT#: PROJ. AGRISTARS
81/05/00 91 PAGES UNCLASSIFIED DOCUMENT

UTTL: A computer program for the simulation of heat and moisture flow in soils
AUTH: A/CAMILLO, P.; B/SCHMUGGE, T. J. PAA: A/(Computer Sciences Corp.,
Silver Spring, Md.)
CORP: National Aeronautics and Space Administration, Goddard Space Flight
Center, Greenbelt, Md. AVAIL. NTIS SAP: HC A05/MF A01
ERTS

TM-82121
SM-G1-04086

A COMPUTER PROGRAM FOR THE SIMULATION
OF HEAT AND MOISTURE FLOW IN SOILS

by
P. Camillo
Computer Sciences Corporation
Silver Spring, MD 20910

T. J. Schmugge
Hydrological Sciences Branch
Goddard Space Flight Center
Greenbelt, MD 20771

May 1981

GODDARD SPACE FLIGHT CENTER
Greenbelt, Maryland

083-27317#

A COMPUTER PROGRAM FOR THE SIMULATION OF HEAT AND MOISTURE FLOW IN SOILS

ABSTRACT

This document describes a computer program that simulates the flow of heat and moisture in soils. The space-time dependence of temperature and moisture content is described by a set of diffusion-type partial differential equations. The simulator uses a predictor/corrector to numerically integrate them, giving wetness and temperature profiles as a function of time. The simulator was used to generate solutions to diffusion-type partial differential equations for which analytical solutions are known. These equations include both constant and variable diffusivities, and both flux and constant concentration boundary conditions. In all cases, the simulated and analytic solutions agreed to within the error bounds which were imposed on the integrator. Simulations of heat and moisture flow under actual field conditions were also performed. Ground truth data were used for the boundary conditions and soil transport properties. The qualitative agreement between simulated and measured profiles is an indication that the model equations are reasonably accurate representations of the physical processes involved.

A COMPUTER PROGRAM FOR THE SIMULATION OF HEAT AND MOISTURE FLOW IN SOILS

SECTION 1 – INTRODUCTION

It is not generally feasible to attempt an exact simulation of heat and moisture flow in soils under field conditions. The number of variables involved is large, and it is difficult to include all relevant transport processes. However, simulations of this type are useful when used as a laboratory to assess the relative importance of the various factors contributing to the heat and moisture fluxes. They can also be used to predict the changes in heat and moisture profiles caused by the imposition of particular boundary conditions or by modifying the thermal and hydraulic properties of the soil-water system. For example, this simulator will be used as part of a study to assess the utility of radiometric measurements of soil microwave emissions as an indication of the moisture profile below the surface (Schmugge, 1978). Emission models that compute the brightness temperature as a function of soil moisture and temperature profiles are currently being evaluated. To test the emission response to varying boundary conditions (rainfall, water table height, surface heat fluxes, etc.), ground truth data must be measured under the appropriate conditions to use as input to these algorithms, which is not always possible. However, in the initial stages of model assessment, the output of this simulator can be used in place of the ground truth data as it correctly models (at least qualitatively) the response of the profiles to the boundary conditions.

Section 2 contains a mathematical description of the program. As an example of the kind of system this simulator is designed to model, Section 2.1 presents the equations that describe a particular diffusion model of the transport properties. Section 2.2 is an overview of the predictor-corrector method used to numerically integrate the equations. A derivation of the equations used to perform the time integration is given in Section 2.3.

Section 3 describes some test results. The simulator has been used to solve diffusion equations that conform to Fick's law and for which analytical solutions are known. Section 3.1 reports on

these results. In Section 3.2, the results of a day-long field simulation are compared to the ground truth data. The resulting agreement indicates that the simulator solves the model equations correctly.

Section 4 is a complete description of the software, including a baseline diagram, all required input, all output, and the job control language (JCL) required for execution.

SECTION 2 – MATHEMATICAL DESCRIPTION

2.1 DERIVATION OF EQUATIONS

The slow movement of heat and moisture in a porous medium such as soil can be described by diffusion-type equations (Nielson et. al., 1972). In the classical diffusion theory, the flux (the amount of substance crossing a unit area per unit time) is proportional to the negative of the gradient of the concentration. The proportionality factor is the diffusion coefficient. The best known example of this kind of flow is embodied in Darcy's Law (Hillel, 1971). The wetness flux is

$$\vec{q}_\theta = -K(\theta) \vec{\nabla}(\psi(\theta) - z) \quad (2-1)$$

where q_θ is the flux (cubic centimeters of water per square centimeter per second, cm/sec), $K(\theta)$ is the hydraulic conductivity (cm/sec), $\psi(\theta)$ is the matric potential (cm), and z is the distance from some reference point. The term $\psi - z$ is the hydraulic head and is the potential energy of the soil water (matric plus gravitational) per unit weight of water. The function ψ is called the matric potential and is the energy per unit weight required to overcome the capillary and adhesion forces that bind the water in the soil. Because work must be done to remove water from an unsaturated soil, ψ is negative. The distance z is the gravitational potential energy per unit weight.

K and ψ are functions of volumetric wetness θ (cm³ water/cm³ medium). In this application, it is assumed that soil properties change only with depth; thus the gradient is a derivative with respect to z . Therefore, Equation (2-1) may be written as

$$q_\theta = -K(\theta) \left(\frac{d\psi}{dz} - 1 \right) \quad (2-2a)$$

$$= -K(\theta) \frac{d\psi}{d\theta} \frac{d\theta}{dz} + K(\theta) \quad (2-2b)$$

The second line follows from the chain rule of differentiation.

Defining a diffusion coefficient

$$D(\theta) = K(\theta) \frac{d\psi}{d\theta} \quad (2-3)$$

yields, when inserted into Equation (2-2),

$$q_{\theta} = -D(\theta) \frac{d\theta}{dz} + K(\theta) \quad (2-4)$$

The first term in Equation (2-4) is the diffusion contribution to the moisture flux due to a wetness gradient.

There is a large body of experimental evidence indicating that thermal gradients induce moisture flow (Nielsen et. al., 1972). For example, if a uniformly moisture soil sample is enclosed in a horizontal cylinder and is subjected to a thermal gradient, moisture flows from the warm toward the cool end. As field soil temperatures are always changing, an isothermal model such as Equation (2-4) is not complete; a theory that treats both heat and moisture flow in soils is necessary. In the following description, diffusion-type expressions for both heat and moisture fluxes are presented. The derivation closely follows the work of Philip and De Vries (1957). Contributions to heat and wetness fluxes that are proportional to wetness and temperature gradients are described. The conservation of mass and energy is then invoked to derive the partial differential equations that describe the variation with time of temperature and moisture profiles.

The diffusive flux of water vapor in a porous medium is modeled as

$$\vec{q}_v = -D_{atm} f(\epsilon, \theta) \vec{\nabla} \rho \quad (2-5)$$

where \vec{q}_v = vapor flux density (gm/cm²/sec)

D_{atm} = molecular diffusivity of water vapor in air (cm²/sec)

f = tortuosity and porosity function

ρ = density of water vapor (gm/cm³)

Equation (2-5), with $f = 1$, describes the diffusion of water vapor in air (Eagleson, 1970). The factor f represents the reduced volume available for vapor diffusion due to the soil and water and the obstacles to the diffusing substance presented by the soil matrix. An experimentally determined graph of f as a function of θ can be parameterized by a linear function (Jackson et al., 1974).

$$f(\theta) = \alpha (\epsilon - \theta) \quad (2-6a)$$

where ϵ = soil porosity

α = constant less than 1

The diffusivity D_{atm} is a function of temperature and can be adequately modeled by the equation (Kimball et. al., 1976)

$$D_{\text{atm}} \cong D_0 \left(\frac{T}{273.16} \right)^{1.75} \quad (2-6b)$$

where $D_0 = 0.229 \text{ (cm}^2/\text{sec)}$ and T is the absolute temperature.

The gradient in Equation (2-5) is to be evaluated in terms of moisture and temperature gradients as these are the dependent variables of the model. This can be accomplished by using the relationship between vapor density and relative humidity:

$$\rho = \rho_0 h = \rho_0 \exp [(\psi g)/(RT)] \quad (2-7a)$$

where ρ_0 = density of saturated water vapor

h = relative humidity

g = gravitational acceleration constant

R = gas constant for water vapor = $4.615 \times 10^6 \text{ (ergs/gm/degrees Kelvin (}^\circ\text{K))}$

The vapor density ρ_0 depends on temperature and can be approximated by (Kimball et. al., 1976)

$$\rho_0(T) \cong \exp (R_0 - (R_1/T)) \quad (2-7b)$$

where $R_0 = 6.0035$

$R_1 = 4975.9 \text{ (}^\circ\text{K)}$

T = temperature ($^\circ\text{K}$)

Equation (2-7) can be derived from the laws of thermodynamics. Assuming water vapor behaves as an ideal gas, an expression can be readily obtained relating the vapor pressure, the temperature, and the chemical potential of the gas. The chemical potential and the matric potential of liquid water are related because they both represent the free energy of the respective phases and the two phases are in equilibrium. The gas density is proportional to the partial pressure.

The gradient in Equation (2-5) can be expressed in terms of temperature and moisture gradients as follows:

$$\begin{aligned}\vec{\nabla} \rho &= \vec{\nabla} (\rho_0 h) = \rho_0 \vec{\nabla} h + h \vec{\nabla} \rho_0 \\ &= \rho_0 \left(\frac{\partial h}{\partial T} \vec{\nabla} T + \frac{\partial h}{\partial \theta} \vec{\nabla} \theta \right) + h \left(\frac{\partial \rho_0}{\partial T} \vec{\nabla} T + \frac{\partial \rho_0}{\partial \theta} \vec{\nabla} \theta \right)\end{aligned}\quad (2-8)$$

The derivative of h with respect to T can be computed from Equation (2-7):

$$\begin{aligned}\left(\frac{1}{h} \right) \frac{\partial h}{\partial T} &= \frac{-\psi g}{RT^2} + \left(\frac{g}{RT} \right) \left(\frac{\partial \psi}{\partial T} \right) \\ &= \frac{-\ln h}{T} + \ln h \left(\frac{1}{\psi} \right) \left(\frac{\partial \psi}{\partial T} \right)\end{aligned}\quad (2-9)$$

The matric potential is dependent on temperature through the surface tension of water, which is responsible for the capillary force that binds the water to the soil matrix. Therefore, ψ is proportional to surface tension σ (Philip and De Vries, 1957) and

$$\left(\frac{1}{\psi} \right) \frac{\partial \psi}{\partial T} = \left(\frac{1}{\sigma} \right) \frac{\partial \sigma}{\partial T} \quad (2-10)$$

A table giving surface tension at a pressure of one atmosphere as a function of temperature (Eagleson, 1970) can be fit with the exponential

$$\sigma(T) = \sigma_0 \exp [-\gamma(T - 273.16)] \quad (2-11)$$

where $\sigma_0 = 75.9$ dynes (dyn)/cm

$$\gamma = 2.09 \times 10^{-3} \text{ (1/}^\circ\text{C)}$$

T = temperature, $^\circ\text{K}$

The derivative of ψ with respect to T can be computed using Equations (2-10) and (2-11). Equation (2-9) therefore is

$$\frac{dh}{dT} = -h \ln h \left(\gamma + \frac{1}{T} \right) \quad (2-12)$$

The θ dependence of h is, from Equation (2-7),

$$\frac{dh}{d\theta} = \left(\frac{g}{RT} \right) \frac{d\psi}{d\theta} = h \ln h \frac{d}{d\theta} \ln \psi \quad (2-13)$$

Matric potential ψ typically changes by four to six orders of magnitude over the range of wetness values normally found in unsaturated soils. A comparison of Equations (2-12) and (2-13) shows that the variation of h with θ is much larger than the variation of h with T , at least over the range of temperatures found in soils (273 to 310°K). Therefore, relative humidity h in Equation (2-8) is considered to be only a function of θ .

Since water vapor behaves approximately like an ideal gas, its density depends primarily on pressure and temperature. Therefore, ρ_0 can be assumed to be a function of temperature only, with no dependence on θ . With h depending only on θ and ρ_0 depending only on T , Equation (2-8) becomes

$$\vec{\nabla} \rho = \rho_0 \left(\frac{\partial h}{\partial \theta} \right) \vec{\nabla} \theta + h \left(\frac{\partial \rho_0}{\partial T} \right) \vec{\nabla} T \quad (2-14)$$

Inserting Equation (2-14) into Equation (2-5) and using Equation (2-13) for $dh/d\theta$ yields

$$\vec{q}_v = -D_{T,vap} \vec{\nabla} T - D_{\theta,vap} \vec{\nabla} \theta \quad (2-15a)$$

where

$$D_{T,vap} = D_{atm} \alpha (\epsilon - \theta) h \left(\frac{d\rho_0}{dT} \right) \quad (2-15b)$$

$$D_{\theta,vap} = \frac{D_{atm} \alpha (\epsilon - \theta) \rho_0 g h}{RT} \left(\frac{d\psi}{d\theta} \right) \quad (2-15c)$$

This is the sought-for diffusion expression for the vapor flux. Diffusion co-efficients $D_{T,vap}$ and $D_{\theta,vap}$ (respectively valled the “thermal vapor diffusivity” and the “wetness vapor diffusivity”) depend on both θ and T .

The liquid flux can be computed from Darcy’s Law (Equation (2-1)). The gradient of ψ in terms of moisture and temperature gradients is

$$\vec{\nabla} \psi = \frac{\partial \psi}{\partial \theta} \vec{\nabla} \theta + \frac{\partial \psi}{\partial T} \vec{\nabla} T \quad (2-16)$$

Equations (2-10) and (2-11) give the derivative of ψ with respect to temperature:

$$\frac{\partial \psi}{\partial T} = -\gamma \psi \quad (2-17)$$

Thus, the liquid flux is

$$q_l = K \vec{\nabla} z - D_{\theta, liq} \vec{\nabla} \theta - D_{T, liq} \vec{\nabla} T \quad (2-18a)$$

where

$$D_{\theta, liq} = K \left(\frac{\partial \psi}{\partial \theta} \right) \quad (2-18b)$$

$$D_{T, liq} = -K \gamma \psi \quad (2-18c)$$

The total moisture flux, \vec{q}_θ , is the sum of the vapor and liquid fluxes:

$$\vec{q}_\theta = \vec{q}_v + \vec{q}_l \text{ (gm/cm}^2\text{/sec)} \quad (2-19)$$

This can be written in a diffusion form by adding Equations (2-15) and (2-18):

$$\vec{q}_\theta = -D_\theta \vec{\nabla} \theta - D_T \vec{\nabla} T + K \vec{\nabla} z \quad (2-20a)$$

where

$$D_\theta \equiv D_{\theta, liq} + D_{\theta, vap} \quad (2-20b)$$

$$D_T \equiv D_{T, liq} + D_{T, vap} \quad (2-20c)$$

The volumetric water content, θ , is the volume of moisture per unit volume of soil. Because the density of water is 1 gm/cm³, θ also represents the mass of water per volume soil, assuming that the water includes the liquid phase plus the gas phase. As θ represents the mass and q_θ is the mass flux, they are related by the continuity equation:

$$\frac{d\theta}{dt} + \vec{\nabla} \cdot \vec{q}_\theta = 0 \quad (2-21)$$

This is a partial differential equation involving θ as a function of depth and time. An analogous diffusion equation can be derived to describe the time dependence of the temperature profile as a function of the soil heat flux.

Fourier's heat flow equation gives the heat flow due to a temperature gradient:

$$\vec{q}_{h,T} = -\lambda \vec{\nabla} T \quad (2-22)$$

where $\vec{q}_{h,T}$ is the temperature-driven heat flux (calories/cm²/sec) and λ is the thermal conductivity of the medium (cal/cm/sec/°K).

To apply this equation to heat transfer in the soil, the effective thermal conductivity of the soil-water-air system must be known. A generally accepted model (De Vries, 1975) gives λ as a weighted average over the thermal conductivity of each soil constituent:

$$\lambda = \frac{f_w \lambda_w + \sum_i k_i f_i \lambda_i + k_a f_a (\lambda_a + \lambda_{vap})}{f_w + \sum_i k_i f_i + k_a f_a} \quad (2-23)$$

where f_w , f_i , and f_a are the volumetric fractions of the liquid, various solids, and the air, respectively. (It should be noted that f_w and θ are the same, and the porosity, ϵ , is equal to $f_w + f_a$.)

The thermal conductivities of each component are λ_w , λ_i , and λ_a . Factors k_i represent the ratio of the average thermal gradient in the i th soil constituent to the average thermal gradient in water.

They also depend on the shape and orientation of the soil grains. For spheroid-shaped particles, the k_i factors are

$$k_i = \frac{2}{3} \left[1 + \left(\frac{\lambda_i}{\lambda_w} - 1 \right) g_i \right]^{-1} + \frac{1}{3} \left[1 + \left(\frac{\lambda_i}{\lambda_w} - 1 \right) (1 - 2g_i) \right]^{-1} \quad (2-24)$$

where g_i is the shape factor and is equal to 1/2 for cylinders of infinite length, 1/3 for spheres, and 0 for disks of infinite radius.

The weight factor k_a for air can be determined from Equation (2-24), with λ_i equal to the thermal conductivity of dry air. The air shape factor g_i in this case has no physical meaning. It is usually treated as a variable function of wetness that must be determined for each soil type. Near saturation, its value is usually accepted to be 1/3.

The latent heat absorbed or emitted by the soil as the wetness changes state between the liquid and vapor phases can be an important cause of temperature fluctuations. This heat can be included in

the heat flux by assuming that the vapor flux carries with it a heat flux due to the latent heat of vaporization that it absorbed from the soil when it evaporated. This heat flux carried by the vapor phase is

$$\vec{q}_{h,v} = L \vec{q}_v$$

where L is the latent heat of vaporization (cal/gm) and \vec{q}_v is the vapor flux (Equation (2-15)).

Both thermal and moisture gradients contribute to \vec{q}_v and therefore contribute to $\vec{q}_{h,v}$. The moisture contribution is computed by inserting the appropriate term from Equation (2-15) into the above equation:

$$\vec{q}_{h,v(\theta)} = L D_{\theta,vap} \vec{\nabla} \theta \quad (2-25)$$

The temperature gradient contribution from Equation (2-15) is included by increasing the apparent thermal conductivity of the air-filled pores, where the vapor phase primarily exists. This vapor has thermal conductivity λ_{vap} and carries heat flux $-\lambda_{vap} \vec{\nabla} T$ according to Fourier's Law, where $\vec{\nabla} T$ is the temperature gradient in the pore. However, this heat flux can also be represented by the thermal term in Equation (2-15) with porosity factor f set equal to 1: $-D_{atm} h (d\rho_0/dT) \vec{\nabla} T$. By equating these two expressions for the same heat flux, the vapor conductivity is found to be

$$\lambda_{vap} = L D_{atm} h \left(\frac{d\rho_0}{dT} \right) \quad (2-26)$$

Therefore, the total heat flux in the soil is

$$\begin{aligned} \vec{q}_h &= \vec{q}_{h,T} + \vec{q}_{h,v(\theta)} \\ &= -\lambda \vec{\nabla} T - L D_{\theta,vap} \vec{\nabla} \theta \end{aligned} \quad (2-27)$$

where λ is given by Equation (2-23) and includes the vapor thermal conductivity.

The total thermal energy per unit volume of medium at temperature T is CT , where C is the volumetric heat capacity (cal/cm³/°K) and T is the absolute temperature. The conservation of heat energy leads to an equation similar to the conservation of mass for water (Equation (2-21));

$$\frac{d(CT)}{dt} + \vec{\nabla} \cdot \vec{q}_h = 0 \quad (2-28)$$

The volumetric heat capacity of the soil is computed as a sum over the capacities of the constituents (De Vries, 1975);

$$C = \sum_i f_i C_i + f_w C_w + f_a C_a \quad (2-29)$$

Fractions f_i , f_w , and f_a are the volumetric contents of solid, water, and air; and C_i , C_w , and C_a are the heat capacities of the solid constituents, water, and air, respectively.

Equations (2-21) and (2-28) describe the time dependence of soil wetness and temperature profiles. In this application, only vertical fluxes are considered; this constitutes a stratified model of the soil. Therefore, the gradient operator can be replaced by a derivative with respect to soil depth. Thus, the moisture and heat fluxes are, from Equations (2-20) and (2-27),

$$q_\theta = -D_\theta \left(\frac{d\theta}{dz} \right) - D_T \left(\frac{dT}{dz} \right) - K \quad (2-30)$$

$$q_h = -\lambda \left(\frac{dT}{dz} \right) - L D_{\theta, \text{vap}} \left(\frac{d\theta}{dz} \right) \quad (2-31)$$

The time derivatives of the moisture and temperature profiles are, from Equations (2-21) and (2-28),

$$\frac{d\theta}{dt} = - \frac{dq_\theta}{dz} \quad (2-32)$$

$$\frac{dT}{dt} = - \frac{1}{C} \left(\frac{dq_h}{dz} \right) \quad (2-33)$$

BOUNDARY CONDITIONS

To solve these equations, boundary conditions must be supplied for both wetness and temperature at the air/soil interface and in the bottom layer of the profile. In principle, either the fluxes q_θ and q_h or the variables θ and T could be specified. In the simulator both heat and moisture fluxes, q_h and q_θ , are computed at the surface. In this way the effects of the environment (i.e., rainfall, evapotranspiration, radiation, etc.) on the profile evolution can be modeled. At the bottom of the profile a mixture of flux and variable boundary conditions are used. One can specify constant wetness, a downward wetness flux equal to the hydraulic conductivity of the bottom layer, or a flux equal to zero. The temperature in the bottom layer is held constant.

When both temperature and moisture profiles are modeled the surface fluxes can be found by the solution of the heat balance equation

$$S = R + LE + H \quad (2-34)$$

All fluxes are positive downward. S is the heat absorbed by the soil, R is the net radiation flux, LE is the evapotranspiration heat flux, and H is the sensible heat. This equation can be written with the temperature at the soil surface as the only unknown variable. After finding the solution, the heat flux q_h at the surface is set equal to S , and the surface moisture flux q_θ is set equal to E .

The heat flux absorbed by the soil can be evaluated by using the discrete form of equation 2-22

$$S = -\lambda_1 \left(\frac{T_1 - T_s}{z_1} \right) \quad (2-35)$$

where λ_1 = thermal conductivity of the surface layer

T_1 = temperature at the center of the first layer

T_s = surface temperature

z_1 = depth to center of first layer

The net radiation R is usually divided into average short and long wavelength components (Eagleson, 1970):

$$R = R_{\text{short}} + R_{\text{long}} \quad (2-36)$$

The short wave component of R is

$$R_{\text{short}} = (1 - A) [1 - (1 - k) N] I_c \quad (2-37)$$

where A = surface short wave albedo

N = fractional cloud cover

k = fraction of radiation transmitted by a completely cloud covered sky

I_c = insolation at Earth's surface for a cloudless sky

An empirical model for I_c is (Eagleson, 1970)

$$I_c = \begin{cases} I_0 \sin \alpha \exp(-.128 n / \sin \alpha), & \alpha > 0 \\ 0, & \alpha < 0 \end{cases} \quad 2-38$$

where I_O = short wave solar energy flux incident at the top of the Earth's atmosphere (.033 cal/cm²/sec)

n = air turbidity factor ($n \sim 2-5$)

α = angle between the sun and the local tangent plane computed from the following:

$$\sin \alpha = \sin \delta \sin \phi + \cos \delta \cos \phi \cos \tau \quad (2-39)$$

δ = angle between Sun and plane of celestial equator ($-23 \text{ deg} \leq \delta \leq +23 \text{ deg}$)

ϕ = local latitude

τ = hour angle of the Sun

$$= W_d (t - 12)$$

t = hour of the day

$$W_d = \pi/12 \text{ (rad/hour)}$$

The contribution to the net radiation from the long wavelength part of the spectrum is modeled by

$$R_{\text{long}} = \sigma E_a T_a^4 - \sigma T_s^4 \quad (2-40)$$

where T_a = air temperature

E_a = emissivity of the air

T_s = surface temperature

σ = Stefan-Boltzmann constant

The air emissivity is modeled as (Eagleson, 1970)

$$E_a = .74 + .005 e_a \quad (2-41)$$

where e_a is the vapor pressure in milli-bars. The vapor pressure and air temperature are supplied by the user, and should be taken from the same height above the surface. Equation 2-40 is a mathematical statement of the assumption that both the air and soil surface radiate with emissivities of E_a and one respectively.

A standard model for the heat carried by the evapotranspirational flux is (Eagleson, 1970)

$$LE = - \frac{\gamma p L \bar{U}_a}{P V^2 \ln^2 \left(\frac{Z}{Z_o} \right)} (e_s - e_a) \quad (2-42)$$

where L = latent heat of vaporization

P = density of air = 1.15×10^{-3} gms/cm³

p = atmospheric pressure in units of e_s and e_a (~ 1000 millibars)

V = Von Karman's constant (2.5)

Z_o = surface roughness parameter ($\sim .025$ cm. for smooth surface)

\bar{U}_a = wind velocity at height Z (cm/sec), averaged over 1 hour

e_a = vapor pressure at height Z

e_s = vapor pressure at soil surface

γ = psychrometric constant (0.61808 mb/ $^{\circ}$ K)

The surface vapor pressure is computed from

$$e_s = h e_{sat}$$

e_{sat} = saturation vapor pressure

$$= \frac{\rho_{sat} (T_s) R_{gas} T_s}{1000} \text{ mb} \quad (2-43)$$

R_{gas} = gas constant

ρ_{sat} = density of saturated water vapor (Equation 2-7b)

h = relative humidity of surface soil water (Equation 2-7a)

The evapotranspiration model outlined above has proven to be most accurate for computing fluxes which have been averaged over a time period on the order of an hour. However, in this simulator the model is used to calculate instantaneous fluxes. The comparison between simulated and measured evaporation rates in Section 3 shows that this approach gives qualitatively correct rates.

The mathematical model of Equation 2-42 is implemented as follows:

$$LE = -C_o - C_1 \bar{U}_a (e_s - e_a) \quad (2-44)$$

The user supplies the constants C_o , C_1 , and \bar{U}_a and e_a as functions of time. C_o would normally have the value zero. To input a constant rate offset in cm/sec, C_o would equal the rate times the latent heat of vaporization (586 cal/cm³). By comparing equations 2-42 and 2-44, it can be seen that C_1 is equal to the following expression:

$$C_1 = \frac{\gamma \rho L}{PV^2 \ln^2 \left(\frac{Z}{Z_o} \right)}$$

The sensible heat flux H in Equation 2-34 is calculated by the Bowen ratio method. This ratio, β , is the ratio of sensible heat to the evapotranspiration flux. Under the assumption that the turbulent transfer coefficients for the two processes are equal, this ratio is

$$\beta = \frac{H}{LE} = \gamma \left(\frac{T_s - T_a}{e_s - e_a} \right) \quad (2-45)$$

In the simulator the Bowen ratio is computed from Equation 2-45, and the sensible heat flux is then calculated using

$$H = \beta LE \quad (2-46)$$

The terms of the heat balance equation are all functions of the unknown surface temperature T_s and the known meteorological variables e_a , \bar{U}_a , and T_a . The method for specifying the values of air vapor pressure, wind speed, and air temperature is described in Section 4.

To solve the heat balance equation for T_s , one starts by rewriting Equation 2-35 as (Hillel, 1977)

$$T_s = \frac{Z_1}{\lambda_1} S + T_1$$

Inserting the right hand side of the heat balance equation (2-34) for S gives

$$T_s = T_1 + \frac{Z_1}{\lambda_1} \left(R(T_s) + LE(T_s) + H(T_s) \right) \quad (2-47)$$

The dependence of the flux terms on T_s has been explicitly noted here. This equation is of the form

$$T_s = F(T_s) \quad (2-48)$$

and can be solved by the method of successive of approximations. A trial value for T_s is chosen, $F(T_s)$ is evaluated, and a new value for T_s results. This procedure can be repeated until satisfactory convergence is obtained. In the simulator the air temperature T_a is used as a start value, and a

maximum of five iterations are allowed. Tests have shown that the process converges after one or two iterations. Convergence is defined as the absolute value of the change in T_s between iterations being less than 0.1°K . The process will always converge if the magnitude of the derivative of F is less than 1,

$$\left| \frac{dF}{dT_s} \right| < 1$$

This derivative is a complicated function of changing meteorological variables, so an analytical study of the conditions required for convergence is not feasible. However from Equation 2-47 it is clear that this derivative is proportional to Z_1 , the depth to the center of the first layer. If the iterative procedure does not converge, then the program should be executed with a thinner surface layer. Lack of convergence will be evident because T_s will be unphysical, either too large or too small.

Periods of rainfall can also be modeled. The user supplies the number of rain storms, the start and stop times of the rain (t_o and t_1), and the total accumulation (r_{tot}) for each one. A constant rate throughout each time interval is assumed and calculated as

$$r = \frac{r_{tot}}{t_1 - t_o} \quad (2-49)$$

The short wave attenuation factor during each rain storm must also be supplied. This number is equivalent to the cloud attenuation factor $1 - (1-k)N$ in Equation 2-37.

During periods of rain, the evapotranspiration and sensible heat fluxes are set equal to zero (LE and H , Equation 2-34), and the wetness flux at the surface q_o , is set equal to the rain rate, Equation 2-49.

It is possible to remove all temperature dependence from the simulation. (See the description of the NAMELIST parameter ITEMPS). In this case the temperature profile, soil heat fluxes, and atmospheric heat fluxes are not modeled. This simulation model is useful when studying moisture flow in relatively moist soils. In this case the temperature gradient contribution to the moisture

flux is negligible, and execution time is significantly decreased when the temperature profile is not modeled. However, the evapotranspiration flux must still be estimated. To do this a Gaussian function of time is supplied,

$$E(t) = -E_{\max} \exp [-k (t - t_{\max})^2] \quad (2-50)$$

where t_{\max} is the time of maximum demand and E_{\max} is the rate at this time. The variable k which determines the width of the Gaussian can be related to the integrated daily rate E_{day} as follows:

$$\begin{aligned} E_{\text{day}} &= \int_{\text{day}} dt |E(t)| \cong E_{\max} \int_{-\infty}^{\infty} dt \exp [-k(t - t_{\max})^2] \\ &= E_{\max} \sqrt{\frac{\pi}{k}} \end{aligned} \quad (2-51)$$

This gives

$$k = \pi \left(\frac{E_{\max}}{E_{\text{day}}} \right)^2 \quad (2-52)$$

The user supplies t_{\max} , E_{\max} , and E_{day} . The simulator computes k from Equation 2-52, and then Equation 2-51 is used to model the evapotranspiration flux.

For some simulations it may be simpler to specify the integrated daily total and the width of the Gaussian. The maximum rate E_{\max} can be computed from these two. The exponential slope k equals $1/t_e^2$, where t_e is the time interval between the maximum rate and the time when the rate falls to $1/e$ of this value. Setting equation 2-52 equal to $1/t_e^2$ and solving for E_{\max} gives

$$E_{\max} = \frac{1}{\sqrt{\pi}} \frac{E_{\text{day}}}{t_e}$$

Therefore the user can compute the value of the required input E_{\max} from E_{day} and t_e .

Figure 2-1 compares the model of equation 2-50 to measured data. The evaporation rates were measured during a field experiment performed in Phoenix, Arizona in 1971 (Jackson, R.D.). Section 3.2 of this document discusses this data in more detail. Curves a and b represent evaporation

rates three and ten days after irrigation. The squares and circles are the data, and the solid lines represent the model of equation 2-50. The maximum rate E_{max} and associated time t_{max} , and the daily total E_{day} used in evaluating equation 2-50 were estimated from the data.

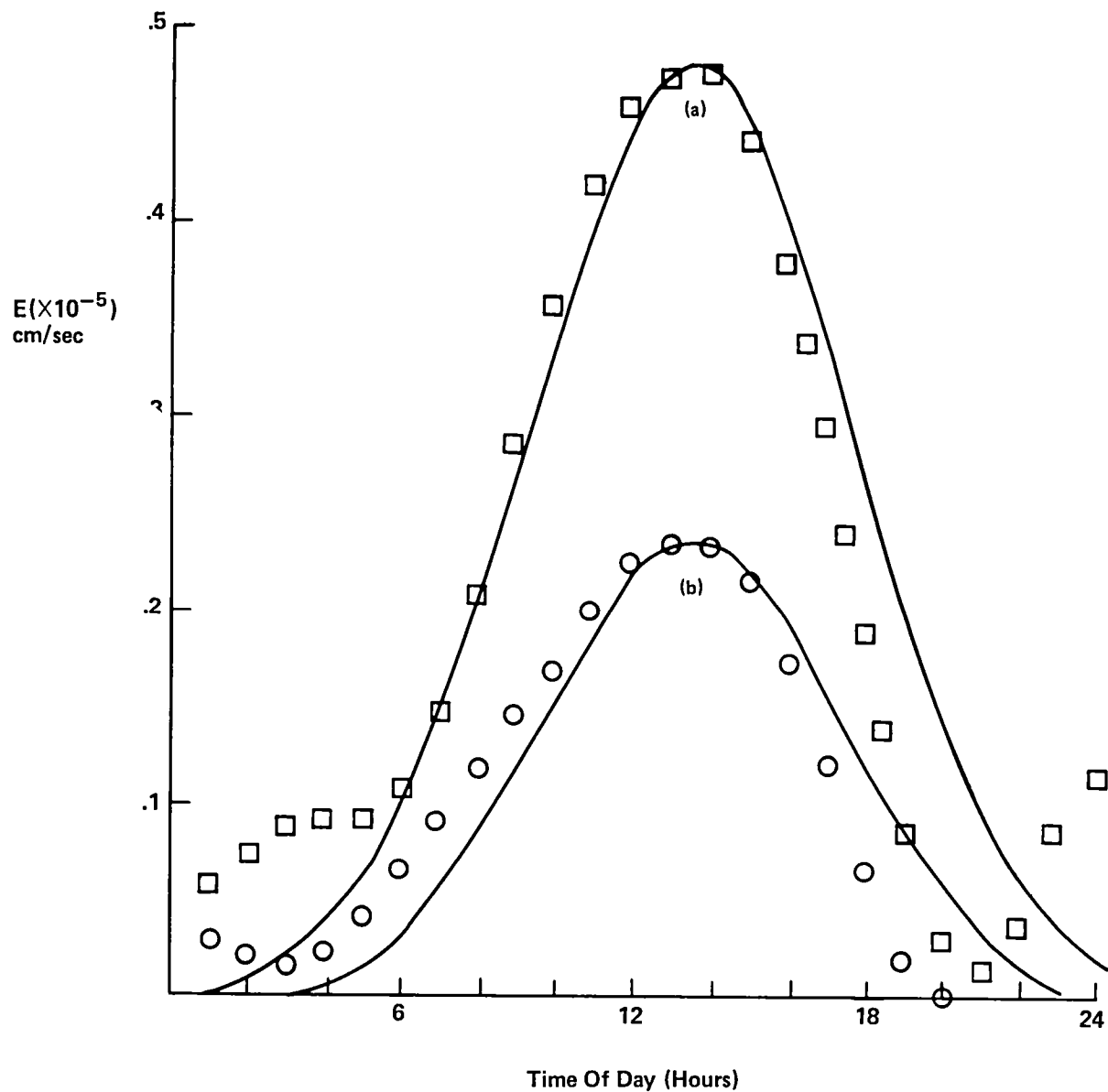


Figure 2-1. Comparison of measured and modelled evaporation rates. The solid lines are the model of equation 2-50, and the squares and circles are the data. Curves a and b represent evaporation three and ten days after irrigation, respectively.

Figure 2-1 shows that equation 2-50 provides a qualitatively acceptable representation of evapotranspiration during daylight hours, when the process is most important. The model does not include the rise in the data from sunset to sunrise. Also the data points are not exactly symmetrical about the maximum.

It is also possible to model the surface temperature and the heat balance equation (and thereby include the effect of the meteorological variables on evapotranspiration) without modeling the soil temperature profile. The surface temperature T_s and average subsurface temperature \bar{T} are modeled by the force restore method (Lin, 1980). The mathematical model is

$$\frac{dT_s}{dt} = \frac{2S}{a} - \frac{2\pi}{\tau} (T_s - \bar{T}) \quad (2-53a)$$

$$\frac{d\bar{T}}{dt} = \frac{S}{a\sqrt{365\pi}} \quad (2-53b)$$

where S is the heat flux absorbed by the soil and

$$a \equiv \sqrt{\frac{\lambda c \tau}{\pi}}$$

In this expression λ is the thermal conductivity of the surface layer, c is the heat capacity, and τ is the number of seconds in a day. The thermal conductivity and heat capacity are computed using Equations 2-23 and 2-29.

Since Equation 2-53 gives the time dependence of the surface temperature explicitly, T_s is made one element of the state vector and is therefore known. Therefore, no iteration is required to solve the heat balance equation. The terms R , LE , and H , are evaluated using the state vector value for T_s , and S is computed from Equation 2-34.

ROOT MODEL

A model of soil water depletion by plant roots has been included as an extra term in the equation of continuity, Equation 2-32:

$$\frac{d\theta}{dt} = -\frac{dq}{dz} - Q(z,t) \quad (2-54)$$

The sink term Q (1/sec) in Equation 2-54 is positive when water flows from the soil to the plant.

The mathematical model is (Hillel, 1977; Gardner, 1964)

$$Q(z,t) = \frac{\Phi_s(\theta,z) - \Phi_p(t)}{\Omega_s + \Omega_p} \quad (2-55)$$

where $\Phi_s(\theta,z)$ is the total potential energy of the soil water;

$$\Phi_s(\theta,z) = \psi(\theta) - z \quad (2-56)$$

where $\psi(\theta)$ = matric potential

$-z$ = gravitational potential (z is the depth below the surface and is positive)

The plant potential Φ_p (cm) in Equation 2-55 varies with time but is assigned the same value throughout the root system. The soil resistance Ω_s (cm-sec) is inversely proportional to the soil conductivity and the amount of active roots:

$$\Omega_s = 1/(B K(\theta) P(z)) \quad (2-57)$$

where B = constant

$K(\theta)$ = hydraulic conductivity (cm/sec)

$P(z)$ = relative root density (1/cm²)

The resistance to flow in the roots Ω_p is also modeled as inversely proportional to the root density and root conductivity. The inverse of conductivity, called the specific resistance, is sometimes used. The plant resistance is

$$\Omega_p(z) = r/P(z) \quad (2-58)$$

where r = specific resistance to flow in the roots (sec/cm).

Using Equations 2-57 and 2-58 for the resistances and Equation 2-56 for the soil water potential energy in Equation 2-55 and rearranging gives

$$Q = \frac{BKP[\psi - z - \Phi_p]}{1 + BKr} \quad (2-59)$$

$$= \frac{BKP[\psi - z - \Phi_p]}{1 + \Omega_p/\Omega_s}$$

The important model parameters are the relative root density $P(z)$ and the ratio of the resistances Ω_p/Ω_s . No loss of generality results from setting $B = 1$, since its value can be absorbed into the definitions of P and r . Since Q is proportional to P , multiplying P at all depths by a constant would only change the rate at which the moisture profiles evolve. Since P has the dimensions ($1/\text{cm}^2$), it is commonly thought of as the length of active roots per volume of soil. As yet there is no experimental evidence that this is true; the model only requires that $P(z)$ represent the relative ability of the roots to absorb water at each depth. The plant potential Φ_p , commonly referred to as the crown potential, is modeled as a response to an atmospheric evapotranspiration demand function.

The discrete model of the sink term as used in this simulator is

$$Q_j = \frac{K_j P_j [\psi_j - z_j - \Phi_p]}{1 + rK_j} \quad (2-60)$$

Q_j is the value of the sink in the j^{th} soil layer, and z_j is the depth to the center of this layer. K_j and ψ_j are the hydraulic conductivity and matric potential of the soil water in the layer. The relative root density in each layer P_j and the specific resistance of the plant roots r are input parameters.

The crown potential $\Phi_p(t)$ is modeled as a response to a known transpiration demand function $E_{p\ell}(t)$. The crown potential is computed by requiring that the integral of the sink terms over the soil profile be equal to $E_{p\ell}$. In its discrete form, this integral is

$$E_{p\ell} = \sum_{j=1}^N Q_j dz_j \quad (2-61)$$

where dz_j is the thickness of the j^{th} layer and N is the number of layers in the profile model. Using Equation 2-60 for Q_j and solving for the crown potential gives

$$\Phi_p(t) = \frac{E_{p\ell}(t) + \sum_{j=1}^N K_j P_j (\psi_j - z_j) dz_j}{\sum_{j=1}^N K_j P_j dz_j} \quad (2-61)$$

Both $E_{p\ell}$ and ψ_j are negative, so Φ_p is also negative. Its magnitude can be large if either the demand is large, the soil is dry (so $|\psi_j|$ is large), or both. The magnitude of Φ_p must be less than the magnitude of the wilting point, Φ_w , which is the largest potential for water intake the plant can create before wilting. Thus, the crown potential must satisfy the inequality

$$\Phi_w \leq \Phi_p \leq 0 \quad (2-62)$$

If $\Phi_p < \Phi_w$ the simulator will set $\Phi_p = \Phi_w$. Once Φ_p is calculated the sink term can be evaluated for each layer using Equation 2-60. It must be positive for all layers, to correspond to flow from soil to roots. Any of the Q_j which are negative are set equal to zero. This procedure is used to accommodate experimental evidence that water flow from plant roots to the soil is negligible (reference 8).

The transpiration demand E_p is computed from the total evapotranspiration demand $E(t)$. This is known from the solution of the heat balance equation, or from the function of Equation 2-50 when heat fluxes are not modeled. A fraction f of the total demand will be satisfied by soil evaporation, E_s ;

$$E_s = f E \quad (2-63)$$

where typically $f \cong 0.1$ (Eagleson, 1970).

The rest of the demand will be satisfied by plant transpiration $E_{p\ell}$;

$$E_{p\ell}(t) = E - E_s = (1 - f) E(t) \quad (2-64)$$

This is the transpiration demand used in Equation 2-61 to compute the crown potential.

SOIL HYDRAULIC PROPERTIES

Both matric potential and hydraulic conductivity as functions of volumetric wetness can be modelled as follows for a wide range of soil types and textures, (Clapp and Hornberger, 1978);

$$K(\theta) = K_s \left(\frac{\theta}{\theta_s} \right)^{2b+3} \quad (2-65a)$$

$$\psi(\theta) = \psi_s \left(\frac{\theta}{\theta_s} \right)^{-b} \quad (2-65b)$$

where θ_s is the volumetric wetness at saturation, K_s and ψ_s are the conductivity and matric potential respectively at saturation, and b is a parameter determined primarily by the soil texture. Representative values are 4 for sand to 11 for clay. This model has been implemented in the simulator.

2.2 METHOD OF SOLUTION

The solution space for $\theta(z,t)$ and $T(z,t)$ consists of a variable time grid and a fixed space grid. The soil is divided into n layers. These layers need not all have the same thickness. However, they should be small enough to enable θ and T within two adjacent layers at a fixed time to be adequately represented by a linear function of depth. For a soil with n layers, Equations (2-32) and (2-33) become $2n$ partial differential equations:

$$\frac{d\theta_i}{dt} = -\frac{dq_{\theta_i}}{dz} \quad (2-66a)$$

$$q_{\theta_i} = K_i - D_{\theta_i} \left(\frac{d\theta_i}{dz} \right) - D_{T_i} \left(\frac{dT_i}{dz} \right) \quad (2-66b)$$

$$\frac{dT_i}{dt} = -\frac{1}{C_i} \left(\frac{dq_{h_i}}{dz} \right) \quad (2-66c)$$

$$q_{h_i} = -\lambda_i \left(\frac{dT_i}{dz} \right) - L D_{\theta, \text{vap}} \left(\frac{d\theta_i}{dz} \right) \quad (2-66d)$$

If the force-restore method is used, the n equations 2-64c are replaced by the two equations 2-53a and b.

Each equation describes the time dependence of θ or T at a fixed depth. These equations are coupled, as spatial derivatives of θ and T appear on the righthand sides.

An Adams-Bashforth numerical integration scheme is used to perform the time integration. The general equation that can be solved by this method is of the form

$$\frac{d\vec{y}}{dt} = \vec{f}(t, \vec{y}) \quad (2-67)$$

where \vec{y} is called the state vector.

Knowledge of $\vec{f}(t, \vec{y})$ enables the generation of solutions $\vec{y}(t)$. In this application, state vector \vec{y} contains θ and T at various depths within the soil. Functions \vec{f} are the spatial derivatives of the fluxes, the righthand sides of Equations (2-66), and these are known functions of z , t , θ , and T .

The elements of \vec{y} and \vec{f} are

$$y_i = \begin{cases} \theta_i & i = 1, n \\ T_i & i = n + 1, 2n \end{cases} \quad (2-68a)$$

$$f_i = \begin{cases} -\frac{dq_{\theta_i}}{dz} & i = 1, n \\ \left(\frac{-1}{C_i} \right) \left(\frac{dq_{T_i}}{dz} \right) & i = n + 1, 2n \end{cases} \quad (2-68b)$$

If the force-restore method is used, the temperature variables are replaced by the following;

$$\begin{aligned} y_{n+1} &= T_s \\ y_{n+2} &= \bar{T} \end{aligned} \quad (2-68c)$$

The corresponding time derivatives f_{n+1} and f_{n+2} are given by equations 2-53a and b.

The time integration is performed via the following steps:

1. Estimate $\vec{y}(t + \Delta t)$, using known values $\vec{y}(t)$, $\vec{f}(t)$, $\vec{f}(t - \Delta t)$, $\vec{f}(t - 2\Delta t)$, \dots , $\vec{f}(t - k\Delta t)$.

The estimate is usually called the predicted value $\vec{y}^{(p)}(t + \Delta t)$.

2. Compute $\vec{f}(t + \Delta t)$, using the elements of $\vec{y}^{(p)}(t + \Delta t)$ from step 1 for θ_i and T_i .

3. Recompute $\vec{y}(t + \Delta t)$, using known values $\vec{y}(t)$, $\vec{f}(t + \Delta t)$ (step 2), $\vec{f}(t)$, \dots , $\vec{f}(t - (k-1)\Delta t)$.

This is called the corrected value $\vec{y}^{(c)}(t + \Delta t)$.

In principle, steps 2 and 3 could be iterated until they both give approximately the same answer.

However, it is more economical to reduce the integration step size Δt until convergence is achieved on the first iteration.

The order of this method is determined by k , the number of back values of \vec{f} that are used. A fourth order method is used in this application. The actual equations used to compute $y^{(p)}$ and $y^{(c)}$ are

$$\vec{y}^{(p)}(t + \Delta t) = \vec{y}(t) + \Delta t \left[1 + \frac{1}{2} \nabla \vec{f}(t) + \frac{5}{12} \nabla^2 \vec{f}(t) + \frac{3}{8} \nabla^3 \vec{f}(t) + \frac{251}{720} \nabla^4 \vec{f}(t) \right] \quad (2-69)$$

and

$$\vec{y}^{(c)}(t + \Delta t) = \vec{y}(t) + \frac{\Delta t}{720} [251 \vec{f}(t + \Delta t) + 469 \vec{f}(t) + 109 \nabla \vec{f}(t) + 49 \nabla^2 \vec{f}(t) + 19 \nabla^3 \vec{f}(t)] \quad (2-70)$$

where functions $\nabla^j \vec{f}(t)$ are linear combinations of the back values of \vec{f} , defined by

$$\begin{aligned} \nabla \vec{f}(t) &= \vec{f}(t) - \vec{f}(t - \Delta t) \\ \nabla^2 \vec{f}(t) &= \nabla(\nabla \vec{f}(t)) = \vec{f}(t) - 2\vec{f}(t - \Delta t) + \vec{f}(t - 2\Delta t) \\ \nabla^3 \vec{f}(t) &= \nabla(\nabla^2 \vec{f}(t)) = \vec{f}(t) - 3\vec{f}(t - \Delta t) + 3\vec{f}(t - 2\Delta t) \\ &\quad - \vec{f}(t - 3\Delta t) \\ \nabla^4 \vec{f}(t) &= \nabla(\nabla^3 \vec{f}(t)) = \vec{f}(t) - 4\vec{f}(t - \Delta t) + 6\vec{f}(t - 2\Delta t) \\ &\quad - 4\vec{f}(t - 3\Delta t) + \vec{f}(t - 4\Delta t) \end{aligned} \quad (2-71)$$

These equations are derived in Section 2.3.

To perform the time integration, the functions f_i , which are the spatial derivatives of the wetness and heat fluxes (Equations (2-68)), must be computed.

The soil is divided into n layers, each of thickness dz_i . There are $n+1$ boundaries. The center of each layer is at depth z_i below the surface. The diagram in Figure 2-1 illustrates this division.

First, the fluxes at the $n-1$ interior boundaries $[q_{\theta_i}, q_{T_i}, i = 2, \dots, n]$ are calculated from the known values of θ_i , T_i , and z_i . The fluxes at the two boundaries $[q_{\theta_1}, q_{\theta_{n+1}}, q_{T_1}, q_{T_{n+1}}]$ are then calculated to satisfy the boundary conditions. The surface moisture and heat fluxes q_{θ_1} and q_{T_1} are found from the solution of the heat balance equation. To hold the bottom temperature constant, the net flux into the bottom layer must be zero. The simulator does this by setting $q_{T_{n+1}}$ equal to q_{T_n} . Similarly, if constant wetness in the bottom is to be modelled, then $q_{\theta_{n+1}}$ is set equal to q_{θ_n} . The f_i are then computed using a linear finite difference representation of the derivative:

$$f_i = -\frac{dq_{\theta_i}}{dz} = -\frac{q_{\theta_{i+1}} - q_{\theta_i}}{dz_i} \quad (2-72a)$$

$$f_{i+n} = \frac{1}{C_i} \frac{dq_{h_i}}{dz} = -\frac{1}{C_i} \frac{q_{h_{i+1}} - q_{h_i}}{dz_i} \quad (2-72b)$$

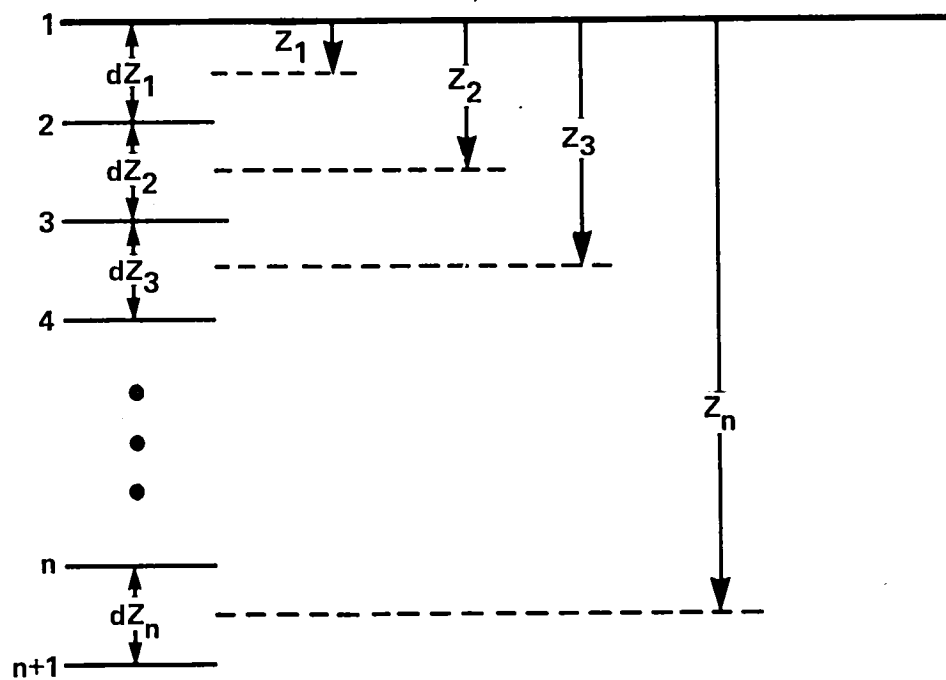
The fluxes at the interior boundaries $[q_{\theta_i}, q_{T_i}, i = 2, \dots, n]$ are computed as follows:

1. In each soil layer, compute K_i (hydraulic conductivity), ψ_i (pressure head), λ_i (thermal conductivity), C_i (volumetric heat capacity), D_{T_i} (thermal diffusivity), and D_{θ, vap_i} (the coefficient of $d\psi/d\theta$ in the vapor contribution to wetness diffusivity; that is, $D_{\theta, \text{vap}} = D'_{\theta, \text{vap}} (d\psi/d\theta)$). The calculation of these quantities as functions of θ_i , Z_i , and T_i are described in Section 2.1.

2. Compare the spatial derivative of the temperature and pressure head at the i th boundary:

$$\left(\frac{d\psi}{dz}\right)_i = \frac{\psi_{i+1} - \psi_i}{z_{i+1} - z_i} = \frac{\psi_{i+1} - \psi_i}{\frac{1}{2}(dz_{i-1} + dz_i)} \quad (2-73a)$$

$$\left(\frac{dT}{dz}\right)_i = \frac{T_{i+1} - T_i}{z_{i+1} - z_i} = \frac{T_{i+1} - T_i}{\frac{1}{2}(dz_{i+1} + dz_i)} \quad (2-73b)$$



NOTE: The center of the i th layer is at depth Z_i .

Figure 2-1. Diagram Showing the Division of the Soil Profile Into n layers, Each One of Thickness dz_i

It should be noted that in Equations (2-66) the spatial derivative of the wetness, $d\theta/dz$, always occurs in the product $(d\psi/d\theta)(d\theta/dz)$. This is equal to $d\psi/dz$ by the chain rule of differentiation; hence, the derivative of ψ , and not θ , is computed. (This means that hysteresis is not included in the simulations).

3. Compute the average value of K , λ , D_T , and $D'_{\theta, \text{vap}}$ at the boundaries by linear interpolation between the values in the two adjacent layers:

$$\overline{K}_i = \frac{K_i dz_{i-1} + K_{i-1} dz_i}{dz_i + dz_{i-1}} \quad (2-74)$$

Similar expressions are used for $\overline{\lambda}_i$, \overline{D}_{T_i} , and $\overline{D}'_{\theta, \text{vap}_i}$.

4. Compute the fluxes. From Equation (2-66), these are

$$q_{\theta_i} = \overline{K}_i - (\overline{K}_i + \overline{D}'_{\theta, \text{vap}_i}) \left(\frac{d\psi}{dz} \right)_i - \overline{D}_{T_i} \left(\frac{dT}{dz} \right)_i \quad (2-75a)$$

$$q_{h_i} = -\overline{\lambda}_i \left(\frac{dT}{dz} \right)_i - L \overline{D}'_{\theta, \text{vap}_i} \left(\frac{d\psi}{dz} \right)_i \quad (2-75b)$$

These fluxes are used to evaluate the righthand sides of Equations (2-72) and thereby perform the time integration.

2.3 DERIVATION OF INTEGRATION EQUATIONS

In the following development, the vector nature of the state vector is ignored to simplify the notation. The references provide a more detailed description. (Teddington, 1958; Booth, 1957; Baginski et. al., 1979)

A general k-step formula to integrate Equation (2-67) can be written in the form

$$Y_{n+1} = h[b_k f_{n+1} + b_{k-1} f_n + \dots + b_0 f_{n+1-k}] + [a_{m-1} Y_n + a_{m-2} Y_{n-1} + \dots + a_0 Y_{n+1-m}] \quad (2-76)$$

where h is the integration step size and choice of coefficients a_j and b_j defines the method. Back values of Y and f plus the current value of f may be used. A multistep method is explicit if $b_k = 0$

and implicit if $b_k \neq 0$. Implicit methods require a recursive evaluation of Y_{n+1} , as f_{n+1} appears in the formula. Despite this drawback, implicit methods are desirable to solve nonlinear problems because they can usually be designed to have smaller truncation errors and better stability. The Adams-Bashforth integrator uses an explicit method, known as a predictor, to calculate Y_{n+1} . f_{n+1} is then calculated using this estimate and an implicit method, known as a corrector, is used to compute a refined value, $Y_{n+1}^{(c)}$. Calculation of derivative f_{n+1} typically involves far more execution time than manipulation of the integrator equations. However, in this formalism f is only computed once per integration step. Therefore, the extra accuracy derived from using the corrector equation costs only a small increase in execution time.

The difference between the corrected and predicted values of Y_{n+1} yields, in most cases, a reliable estimate of the error in the integration. This error estimate is monitored to ensure that the errors are not too large. If necessary, the step size can be reduced by half. If the error is smaller than some lower limit, the step size is automatically doubled, thereby saving execution time.

The predictor/corrector equations define Y_{n+1} in terms of Y_n and derivatives f_{n+1} , f_n , f_{n-1} , etc. (Only one previous value of Y is used.) However, the resulting equation is most conveniently expressed in terms of functions called “backward differences,” which are linear combinations of f_n . The backward difference operator ∇ acting on discrete function g_n is defined by

$$\nabla g_n = g_n - g_{n-1} \quad (2-77a)$$

Higher powers of ∇ acting on g_n are computed by successive applications of ∇ :

$$\nabla^2 g_n = \nabla(\nabla g_n) = \nabla g_n - \nabla g_{n-1} = g_n - 2g_{n-1} + g_{n-2} \quad (2-77b)$$

$$\nabla^3 g_n = \nabla(\nabla^2 g_n) = g_n - 3g_{n-1} + 3g_{n-2} - g_{n-3} \quad (2-77c)$$

$$\nabla^4 g_n = \nabla(\nabla^3 g_n) = g_n - 4g_{n-1} + 6g_{n-2} - 4g_{n-3} + g_{n-4} \quad (2-77d)$$

A useful recursion relation connecting various backward differences at two adjacent time steps is

$$\nabla^k g_{n+1} = \nabla^{k-1} g_{n+1} - \nabla^{k-1} g_n \quad (2-78)$$

Equation (2-76) written in terms of backward differences of f instead of f itself is

$$Y_{n+1} = a_{l-1} Y_n + a_{l-2} Y_{n-1} + \dots + a_0 Y_{n+1-l} + h[c_0 f_n + c_1 \nabla f_n + \dots + c_k \nabla^k f_n + d f_{n+1}] \quad (2-79)$$

To derive the values of coefficients a , c , and d , the backward difference operator ∇ must first be related to derivative operator $D \equiv d/dt$. A Taylor's series extrapolation of a continuous function is

$$Y(t + \Delta t) = \exp(\Delta t D) Y(t)$$

In terms of discrete function g_n , this is

$$g_n = \exp(hD) g_{n-1} \quad (2-80)$$

where h is the step size.

Equation (2-77) can be rewritten as

$$g_n = (1 - \nabla)^{-1} g_{n-1} \quad (2-81)$$

Comparing Equations (2-80) and (2-81) yields

$$\exp(hD) = (1 - \nabla)^{-1} \quad (2-82)$$

or

$$hD = -\ln(1 - \nabla) = \nabla + \frac{\nabla^2}{2} + \frac{\nabla^3}{3} + \dots + \frac{\nabla^n}{n} + \dots \quad (2-83)$$

Equation (2-80) can be used to compute Y_{n+1} given Y_n because the operator hD is known in terms of the backward differences from Equation (2-83):

$$Y_{n+1} = \exp(hD) Y_n \quad (2-84)$$

This can be manipulated as follows:

$$\begin{aligned} Y_{n+1} &= Y_n + (\exp(hD) - 1) Y_n \\ &= Y_n + (\exp(hD) - 1) \left(\frac{hD}{hD} \right) Y_n \end{aligned} \quad (2-85)$$

Because $DY_n = f_n$, this yields

$$Y_{n+1} = Y_n + h \left(\frac{\exp(hD) - 1}{hD} \right) f_n \quad (2-86)$$

The numerator and denominator are expanded in powers of ∇ . From Equation (2-82),

$$\exp(hD) = \frac{1}{1 - \nabla} = 1 + \nabla + \nabla^2 + \dots + \nabla^n + \dots \quad (2-87)$$

Using Equation (2-83) for hD and Equation (2-87) for $\exp(hD)$ in Equation (2-86) yields

$$Y_{n+1} = Y_n + h \left[\frac{\nabla + \nabla^2 + \nabla^3 + \dots + \nabla^n + \dots}{\nabla + \frac{\nabla^2}{2} + \frac{\nabla^3}{3} + \dots + \frac{\nabla^n}{n} + \dots} \right] f_n$$

This division can be carried out to any number of terms. To fourth order, it is

$$Y_{n+1}^{(p)} \cong Y_n + h \left[1 + \frac{1}{2} \nabla + \frac{5}{12} \nabla^2 + \frac{3}{8} \nabla^3 + \frac{251}{720} \nabla^4 \right] f_n \quad (2-88)$$

This is the predictor equation because it does not contain f_{n+1} . The corrector equation can be derived by starting with the identity

$$Y_{n+1} = Y_n + \nabla Y_{n+1}$$

Manipulation as performed above yields

$$\begin{aligned} Y_{n+1} &= Y_n + \nabla \frac{hD}{hD} Y_{n+1} \\ &= Y_n + h \frac{\nabla}{hD} f_{n+1} \end{aligned}$$

Using Equation (2-83) to expand hD gives

$$Y_{n+1} = Y_n + h \left[\frac{\nabla}{\nabla + \frac{\nabla^2}{2} + \dots + \frac{\nabla^n}{n} + \dots} \right] f_{n+1}$$

If the division is carried out to the fourth order, the following corrector equation results:

$$Y_{n+1}^{(c)} \cong Y_n + h \left[1 - \frac{1}{2} \nabla - \frac{1}{12} \nabla^2 - \frac{1}{24} \nabla^3 - \frac{19}{720} \nabla^4 \right] f_{n+1} \quad (2-89)$$

When an integration is performed, the derivative f_{n+1} is known, but the backward differences $\nabla^k f_{n+1}$ are unknown. Therefore, the recursion relation equation (Equation (2-78)) is used to express $Y_{n+1}^{(c)}$ in terms of $\nabla^k f_n$. The resulting corrector equation is

$$Y_{n+1}^{(c)} = Y_n + \left(\frac{251}{720}\right) h f_{n+1} + \frac{h}{720} (469 + 109 \nabla + 49 \nabla^2 + 19 \nabla^3) f_n \quad (2-90)$$

The integrator uses the predictor (Equation (2-88)) to estimate the state at the (n+1) step, and the derivative f_{n+1} is evaluated using this first value. This derivative and previous backward differences are then used to refine the estimate of Y_{n+1} using Equation (2-90). The magnitude of the difference between these two estimates, $|Y_{n+1}^{(p)} - Y_{n+1}^{(c)}|$, is a reliable estimate of the corrector error, and the step size can be changed at this point if necessary. This is determined by testing whether the difference for each state vector element lies within the error window for that layer. This window for the i th layer is the interval

$$(w_l, w_h)_i = (E_i/ED, E_i * ED) \quad (2-91)$$

where E_i and ED are input parameters. If the differences in all layers are less than the lower limit (w_l), the step size is doubled. If the difference in one or more layers is larger than the upper limit (w_h), the step size is halved.

When the integration is complete, the backward difference table is updated via Equation (2-78) to prepare for the next integration step. In addition, if the step size is changed, new backward differences must be computed to reflect the new step size. These can be calculated in terms of the old differences in the following way. The value p is defined to be the ratio of the new interval length to the old length:

$$p = \frac{h_{new}}{h_{old}}$$

Halving gives $p = 1/2$, and doubling results in $p = 2$. In terms of the new interval, the first backward difference is

$$\begin{aligned} \nabla_p f_n &= f_n - f_{n-p} \\ &= f_n - e^{-ph_{old}} f_n \end{aligned}$$

Using Equation (2-82) to express the exponential in terms of ∇ , this becomes

$$\begin{aligned}\nabla_p f_n &= [1 - (1 - \nabla)^p] f_n \\ &= \left[p \nabla - \frac{p(p-1)}{2!} \nabla^2 + \frac{p(p-1)(p-2)}{3!} \nabla^3 \right] \dots f_n\end{aligned}$$

The mth order backward difference is found by raising the first order to the mth power.

$$\nabla_p^m f_n = [1 - (1 - \nabla)^p]^m f_n \quad (2-92)$$

This can easily be expanded in powers of ∇ . Because the new backward differences are linear combinations of the old, changing the step size can simply be effected by a matrix multiplication.

Halving is accomplished as follows:

$$\begin{pmatrix} \nabla \\ \nabla^2 \\ \nabla^3 \\ \nabla^4 \end{pmatrix}_{1/2} = \begin{pmatrix} \frac{1}{2} & \frac{1}{8} & \frac{1}{16} & \frac{5}{128} \\ 0 & \frac{1}{4} & \frac{1}{8} & \frac{5}{64} \\ 0 & 0 & \frac{1}{8} & \frac{3}{32} \\ 0 & 0 & 0 & \frac{1}{16} \end{pmatrix} \begin{pmatrix} \nabla \\ \nabla^2 \\ \nabla^3 \\ \nabla^4 \end{pmatrix} \quad (2-93)$$

and doubling is achieved as shown below.

$$\begin{pmatrix} \nabla \\ \nabla^2 \\ \nabla^3 \\ \nabla^4 \end{pmatrix}_2 = \begin{pmatrix} 2 & -1 & 0 & 0 \\ 0 & 4 & -4 & 1 \\ 0 & 0 & 8 & -12 \\ 0 & 0 & 0 & 16 \end{pmatrix} \begin{pmatrix} \nabla \\ \nabla^2 \\ \nabla^3 \\ \nabla^4 \end{pmatrix} \quad (2-94)$$

Simulator output is desired at predetermined intervals. Because the Adams method integrates by using the largest interval yielding the required accuracy, an integration step frequently ends beyond the desired output time. As a result, interpolation is necessary; however, back values of the state vector are not available for this purpose. A method as accurate as integration and not requiring the storage of previous data points is available (Glang, 1971). This method essentially consists of an equation for integrating a partial step forward or backward using the existing table of backward differences. The derivation closely follows the derivation of the predictor equation. Let

$$p = \frac{t_{n+p} - t_n}{h}$$

Then

$$\begin{aligned} Y_{n+p} &= e^{phD} Y_n \\ &= Y_n + (e^{phD} - 1) Y_n \\ &= Y_n + (e^{phD} - 1) \frac{hD}{hD} Y_n \\ &= Y_n + h \left[\frac{e^{phD} - 1}{hD} \right] f_n \end{aligned}$$

Expressing this as a function of ∇ using Equations (2-82) and (2-83) gives

$$Y_{n+p} = Y_n + h \left[\frac{(1 - \nabla)^{-p} - 1}{-\ln(1 - \nabla)} \right] f_n$$

Expanding in powers of ∇ gives

$$Y_{n+p} = Y_n + h \left[\frac{p + \frac{p(p+1)}{2!} \nabla + \frac{p(p+1)(p+2)}{3!} \nabla^2 + \dots}{\nabla + \frac{\nabla^2}{2} + \frac{\nabla^3}{3} + \dots} \right] f_n \quad (2-95)$$

To the fourth order in ∇ , this is

$$\begin{aligned}
 Y_{n+p} = Y_n + h \left[p + \frac{p^2}{2} \nabla + \left(\frac{2p^3 + 3p^2}{12} \right) \nabla^2 \right. \\
 \left. + \left(\frac{p^4 + 4p^3 + 4p^2}{24} \right) \nabla^3 + \left(\frac{6p^5 + 45p^4 + 110p^3 + 90p^2}{720} \right) \nabla^4 \right] f_n
 \end{aligned}
 \tag{2-96}$$

This is the required interpolation formula.

SECTION 3 – TEST RESULTS

3.1 COMPARISON OF SIMULATED TO ANALYTICAL SOLUTIONS

This program has been tested by using it to solve diffusion-type partial differential equations for which analytical solutions are known. Numerical solutions of the following three equations were generated:

1. Constant diffusion coefficient with constant wetness boundary condition at the surface:

$$\frac{d\theta}{dt} = D_0 \left(\frac{d^2\theta}{dz^2} \right) \quad (3-1)$$

subject to $\theta(z, 0) = \theta_i$ and $\theta(0, t) = \theta_0$. The solution is (Eagleson, 1970)

$$\theta(z, t) = \theta_0 + (\theta_i - \theta_0) \operatorname{erf} \left(\frac{z}{2\sqrt{D_0 t}} \right) \quad (3-2)$$

2. Constant diffusion coefficient with constant flux at the surface:

$$\frac{d\theta}{dt} = D_0 \left(\frac{d^2\theta}{dz^2} \right) \quad (3-3)$$

subject to $\theta(z, 0) = \theta_i$ and $-D_0 (d\theta(0, t)/dz) = f_0$. The solution is (Eagleson, 1970)

$$\theta(z, t) = \theta_i + \frac{2f_0}{D_0} \left[\sqrt{\frac{D_0 t}{\pi}} \exp \left(-\frac{z^2}{4D_0 t} \right) - \frac{|z|}{2} \operatorname{erfc} \left(\frac{|z|}{2\sqrt{D_0 t}} \right) \right] \quad (3-4)$$

3. Concentration-dependent diffusion coefficient with constant wetness at the surface:

$$\frac{d\theta}{dt} = \frac{d}{dz} \left(D(\theta) \frac{d\theta}{dz} \right) \quad (3-5)$$

$$D(\theta) = \frac{D_0}{2} (1 - \ln \theta)$$

subject to $\theta(z, 0) = 0$ and $\theta(0, t) = 1$. The solution is (Philip, 1960)

$$\theta(z, t) = \exp(-z/\sqrt{D_0 t}) \quad (3-6)$$

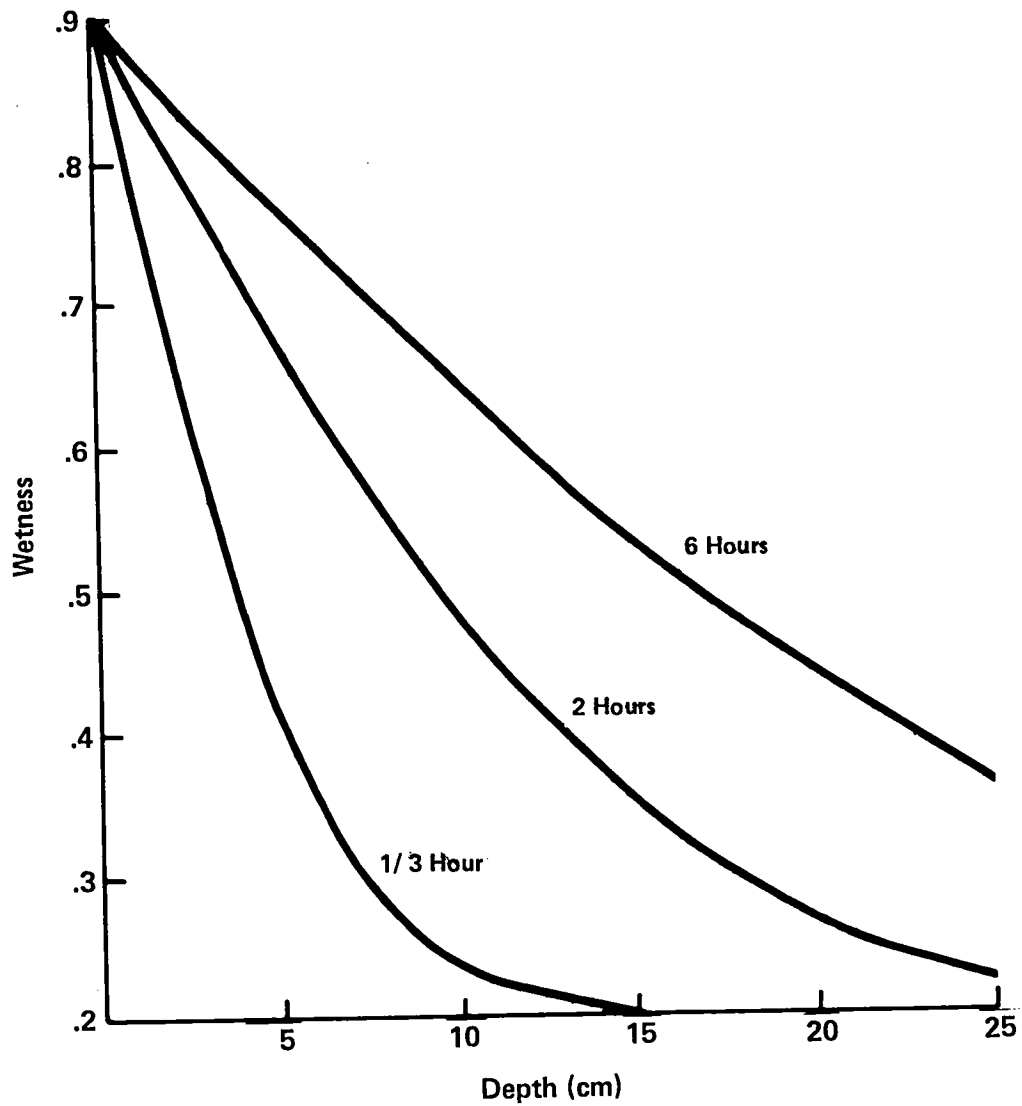
In Equations (3-2) and (3-4), erf and erfc are the error function and complementary error function respectively.

For each simulation, the soil depth was set at 60 cm and the equations were integrated for 6 hours. The diffusion constant, D_0 , was set equal to $0.01 \text{ cm}^2/\text{sec}$. The flux at the bottom was set equal to 0 in all cases. This boundary condition is not compatible with the equations being integrated (as a semi-infinite medium is tacitly assumed in the analytical solution, no bottom boundary condition exists). However, with this value of D_0 , the water incident at the surface could not appreciably infiltrate to a depth of 60 cm in 6 hours; thus, the simulated and exact solutions should agree throughout the top layers. Results are reported to a depth of 25 cm.

The simulator was run with an absolute error tolerance of 0.005. This is the largest allowable magnitude of the difference between predicted and corrected values for each iteration and is a test of convergence. In most cases the difference between the simulated and exact solutions did not exceed 0.001, indicating that the simulation converged to the correct value. The only exception is in the top layers when constant wetness boundary conditions are used (simulations 1 and 3). The exact solution gives constant wetness at $z = 0$, but the simulator requires all layers to have a finite thickness. For these two cases the top two layers were given a thickness of 0.1 cm and the thicknesses of the deeper layers were gradually increased to a maximum of 2 cm. Because of this finite surface thickness, the exact and simulated solutions in the top cm of soil differed by about 0.01. Agreement at other depths was within 0.005.

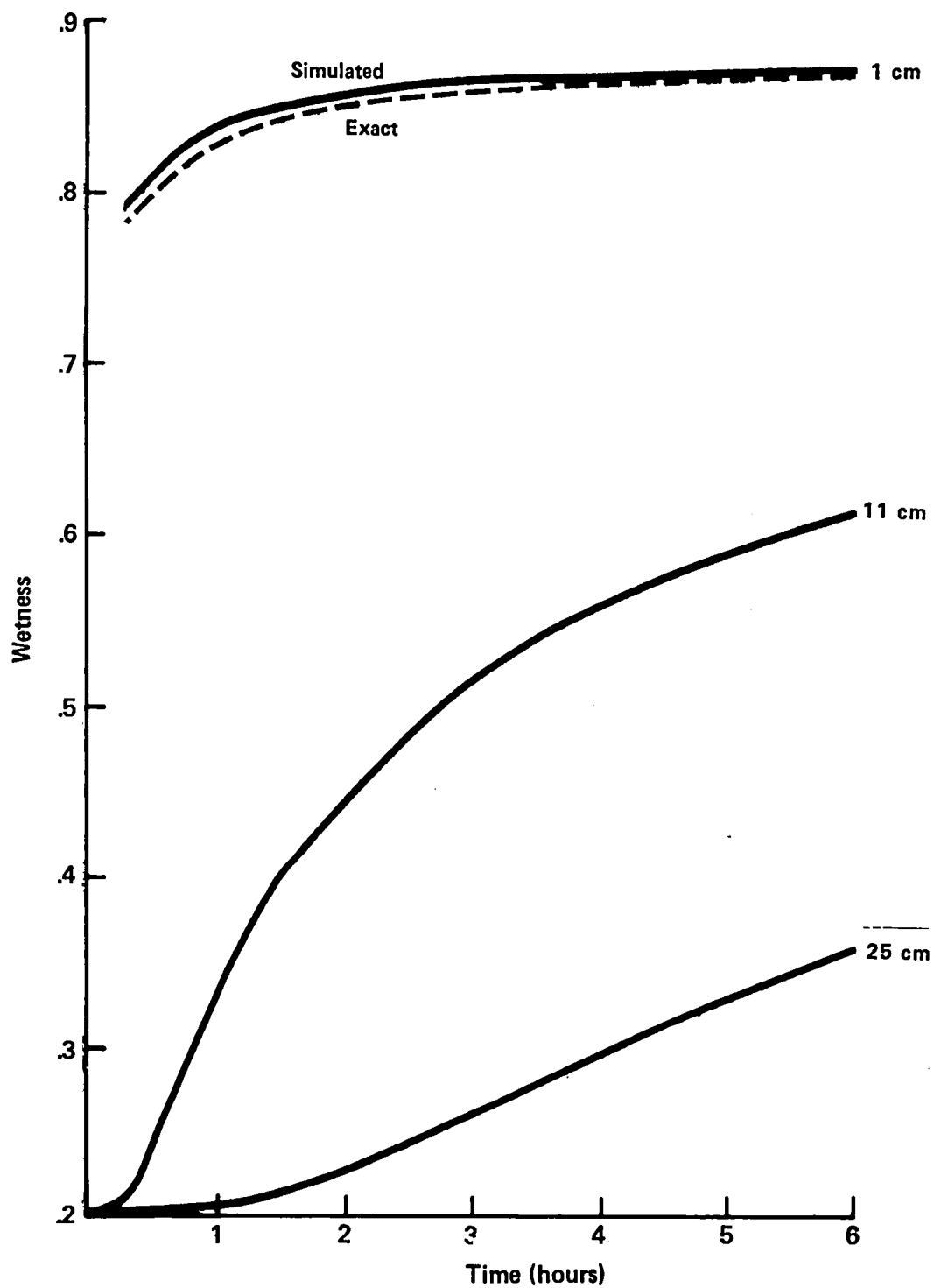
For case 2, the soil was divided into layers 2-cm thick throughout. The exact and simulated solutions were consistently within 0.001.

Figures 3-1 through 3-6 represent the results of the simulation. Figures 3-1, 3-3, and 3-5 give wetness profiles at 20 minutes, 2 hours, and six hours for cases 1, 2, and 3, respectively. Figures 3-2, 3-4, and 3-6 show wetness as a function of time at soil depths of 1, 11, and 25 cm for the three cases.



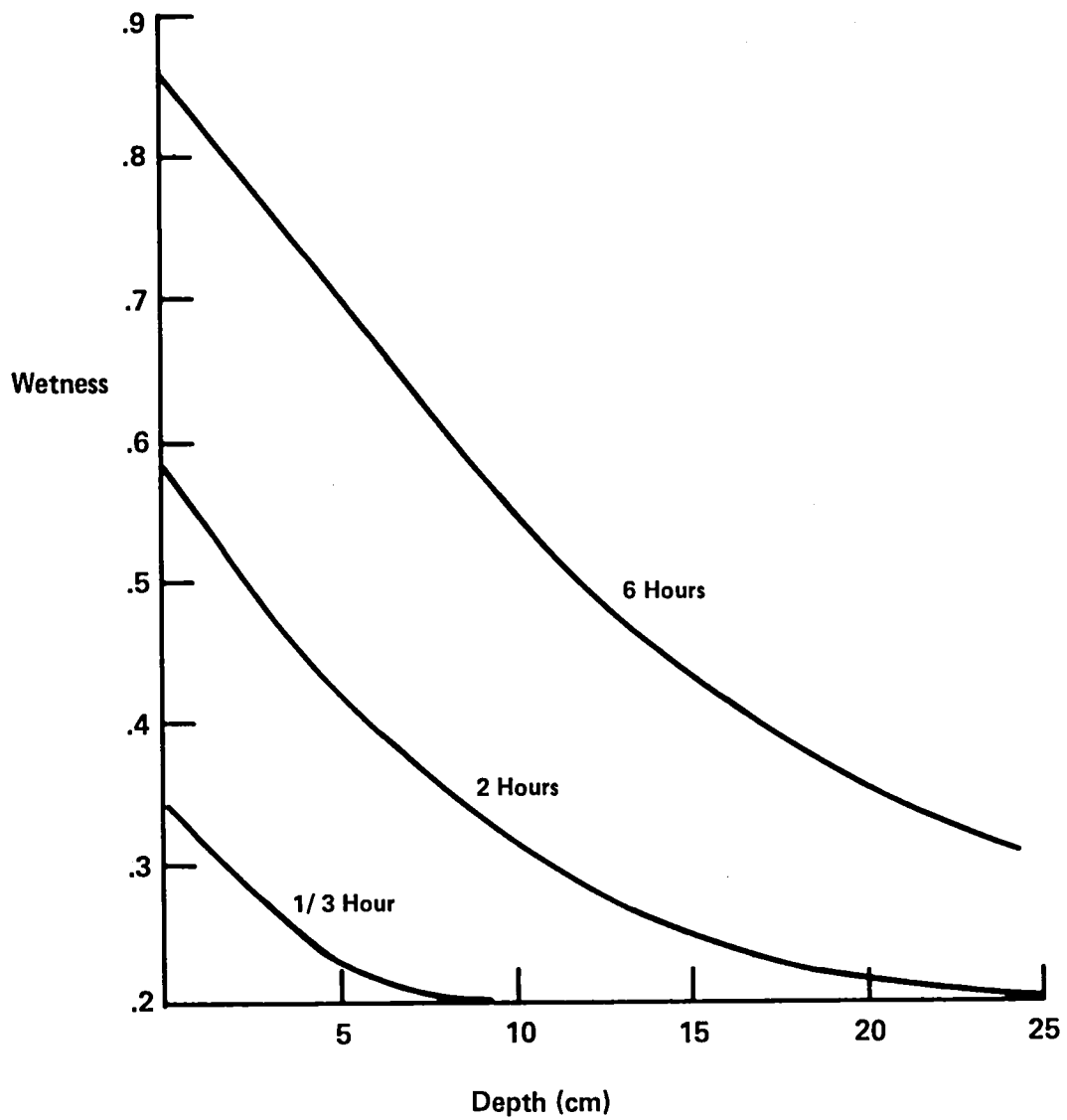
NOTE: Curves are labeled with the simulation time. The initial value of wetness is 0.2 at all depths, and the surface wetness boundary value is 0.9.

Figure 3-1. Wetness Profiles from the Solution of the Diffusion Equation with Constant Diffusivity and Constant Wetness Boundary Conditions (Equations (3-1) and (3-2))



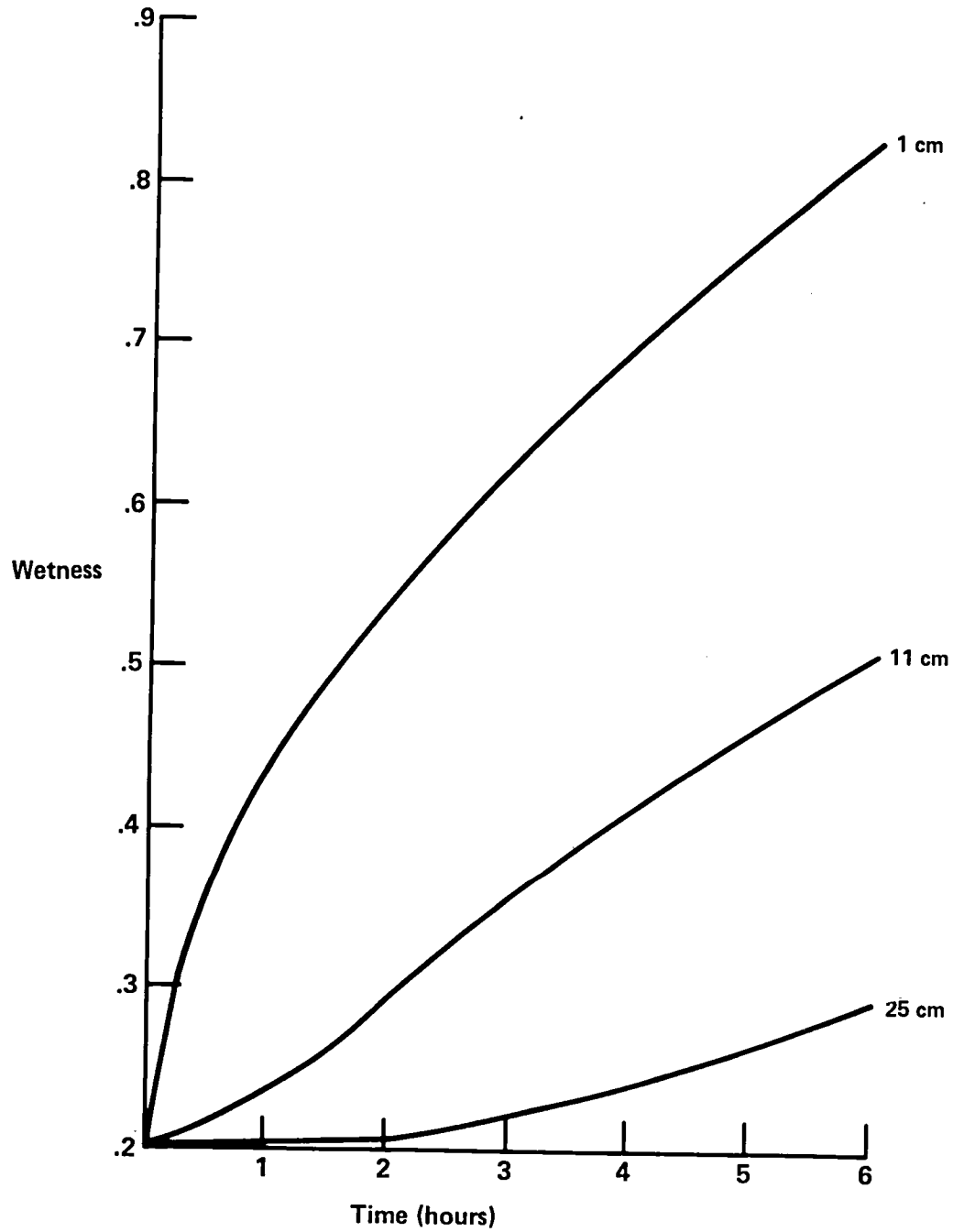
NOTE: The initial and boundary conditions are the same as in Figure 3-1.

Figure 3-2. Wetness as a Function of Time for Three Depths



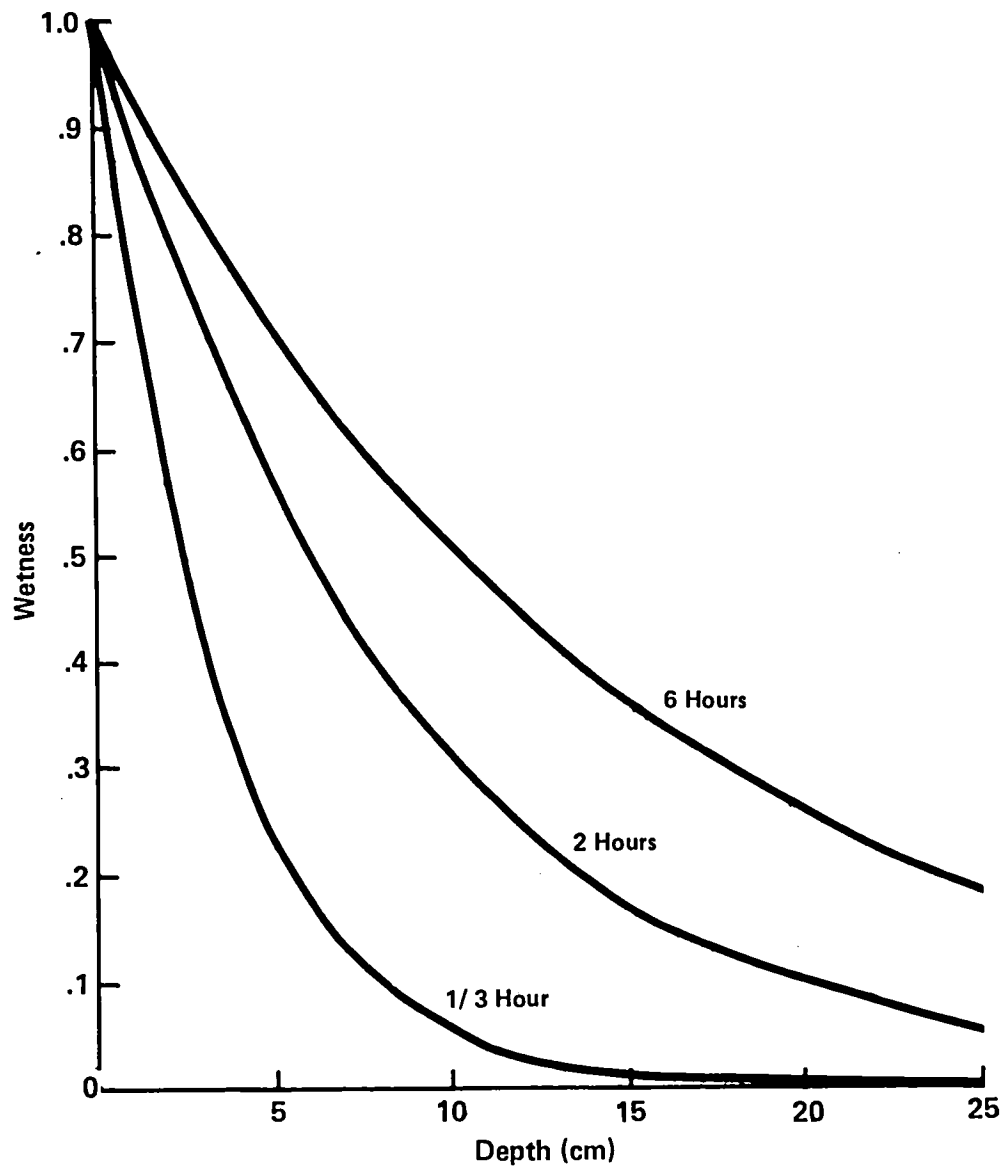
NOTE: Curves are labeled with the simulation time. The initial value of wetness is 0.2 at all depths. The flux at the surface (f_0) has the value 4×10^{-4} (cm/sec).

Figure 3-3. Wetness Profiles From the Solution of the Diffusion Equation With Constant Diffusivity and Constant Boundary Condition (Equations (3-3) and (3-4))



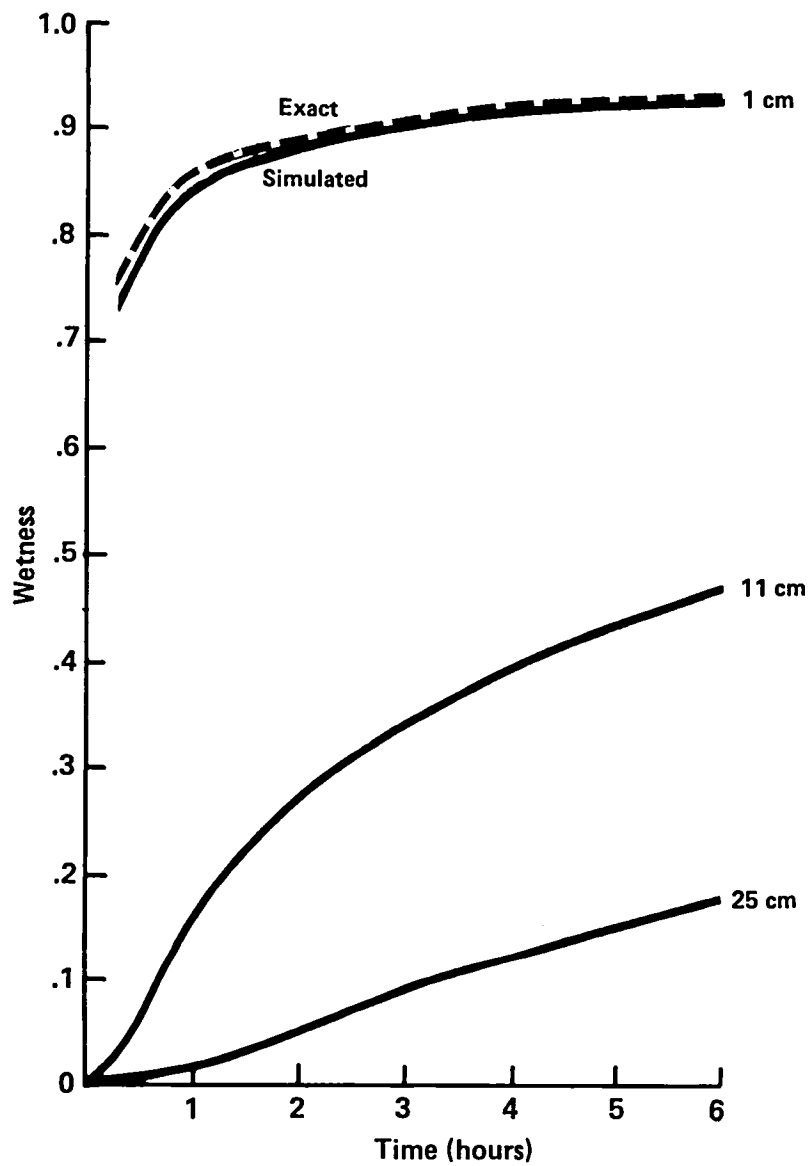
NOTE: The initial and boundary conditions are the same as those in Figure 3-3.

Figure 3-4. Wetness as a Function of Time for Three Depths



NOTE: Curves are labeled with the simulation time. Initially, the profile has a wetness of 0.0 everywhere, and the surface wetness boundary value is 1.0.

Figure 3-5. Wetness Profiles From the Solution of the Diffusion Equation With Variable Diffusivity and Constant Wetness Boundary Condition (Equations (3-5) and (3-6))



NOTE: The initial and boundary conditions are the same as those in Figure 3-5.

Figure 3-6. Wetness as a Function of Time for Three Depths

3.2 FIELD SIMULATION

To gain confidence in the ability of the program to solve the more complicated coupled system of diffusion-type partial differential equations, a day-long simulation of conditions in an actual field was performed. Initial conditions were taken from data that were measured in an experiment performed in March 1971 (Jackson, 1972). In this experiment the soil was irrigated, and at half-hour intervals thereafter for 16 days temperature and moisture profiles, evaporation rates, heat fluxes, and other pertinent meteorological data were measured. Thermal and hydraulic properties of the soil type were also known. The simulation was performed for a 24-hour period starting at the beginning of the fifth day after irrigation.

Subsequent analysis of this data has shown that the flow theory described in Section 2 of this document can provide reasonably accurate moisture fluxes at intermediate values of wetness (Jackson et. al., 1974), but the heat fluxes predicted by this theory are not very accurate (Kimball et al., 1976). Nevertheless, these expressions for moisture and temperature fluxes are generally accepted to be qualitatively correct; therefore, they may be used in a test of the ability of the simulator to qualitatively model heat and moisture flow in soils.

Values for the various input parameters that were used in this simulation are described below.

Semi-log plots of the hydraulic properties of Adelanto loam (the soil used in the experiment) were given in graphical form (Jackson, 1972). These are reproduced here in Figure 3-7, with the scales changed to reflect the different units used in the simulation. Data points were estimated from the curves and then used to find acceptable values for the parameters in the potential and conductivity models, Equations 2-65a and 2-65b. The data points are the logarithms of K and $|\psi|$. The corresponding logarithms of the model equations can be used to solve for the model parameters. To use conductivity and potential data at the same time, the sum of the logarithms of K and ψ was used;

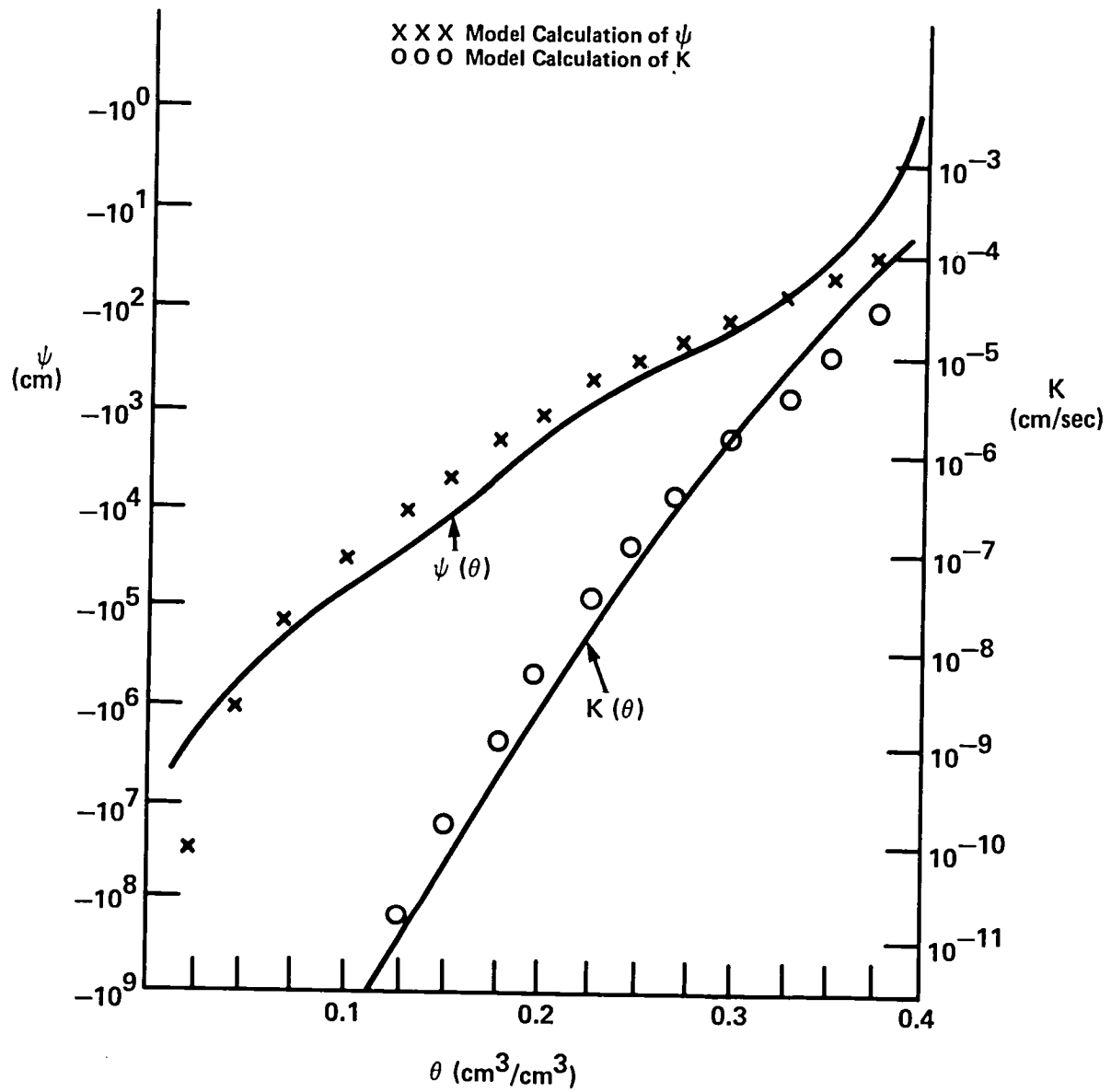


Figure 3-7. Hydraulic properties of Adelanto loam. The solid curves are reproduced from Jackson (1972).

$$\log K(\theta) + \log |\psi(\theta)| = (b + 3) \log \left(\frac{\theta}{\theta_s}\right) + \log |\psi_s| + \log K_s \quad (3-7)$$

The model equations as implemented will not accommodate the sudden change in the slope of the potential curve near saturation. (A procedure to include this behavior is described in Clapp and Hornberger, 1978). Therefore, a value of 0.375 was used for saturation wetness, and calculated matric potentials will not be valid above this value. A linear least squares fit of the data to Equation 3-7 gives a value for the texture parameter b and the sum $\log |\psi_s| + \log K_s$ which is valid for wetness in the range (.125, .375), where both conductivity and potential are known. Values for K_s and ψ_s are then determined by examining the conductivity and potential data separately.

This procedure results in a value of 5.2 for b . However, the resulting fit does not match the matric potential data very well for wetness below 0.1. A slight decrease in the texture parameter will greatly improve the fit for dry conditions, yet does not greatly alter the fit for wetness greater than 0.1. The following values for the model parameters were chosen to match the data:

$$\begin{aligned} b &= 5.0 \\ k_s &= 2.8 \times 10^{-5} \text{ cm/sec} \\ \psi_s &= -42.7 \text{ cm} \\ \theta_s &= 0.375 \end{aligned} \quad (3-8)$$

The points on Figure 3-7 have been calculated with these values in Equation 2-65. The RMS errors of the fit to the logarithms are

$$\sigma_\psi (\log |\psi|) = 0.37 \quad (3-9a)$$

$$\sigma_K (\log |K|) = 0.48 \quad (3-9b)$$

These errors in the logarithms are equivalent to multiplicative errors in the hydraulic properties. With $K_o(\theta)$ and $\psi_o(\theta)$ defined to be the model values for conductivity and potential from Equations 2-65a and 2-65b, the one standard deviation ranges for the error in the fit are $10^{\pm\sigma_K} K_o$ and $10^{\pm\sigma_\psi} \psi_o$. Inserting the values from Equations 3-9a and 3-9b for the RMS errors gives

$$.43 K_o(\theta) \leq K(\theta) \leq 2.3 K_o(\theta) \quad (3-10a)$$

$$.33 \psi_o(\theta) \leq \psi(\theta) \leq 3.0 \psi_o(\theta) \quad (3-10b)$$

These errors are due only to the fit to the model equations. Errors in the measured data are unknown and are therefore not included.

The soil porosity (ϵ , Equation (2-6)) was assumed to be 0.39 because this is the largest value of θ for which the hydraulic properties were given. Also, the maximum values of wetness on the data base for any soil layer at any time is 0.38, which occurred in the top layers just after irrigation.

The volume fractions of the solid constituents of Adelanto Loam are .373 for quartz and .627 for clay (Kimball, et al., 1976). The volume fractions of quartz and clay for the soil-air-water system are therefore

$$\begin{aligned} f_q &= .373 (1-\epsilon) = 0.228 \\ f_c &= .627 (1-\epsilon) = 0.384 \end{aligned} \quad (3-11)$$

The thermal conductivities of quartz and clay are 21 and 7 (mcal/cm/sec/°C) respectively. Equation (2-24) gives, for the weighting factors,

$$\begin{aligned} k_q &= .173 \\ k_c &= .422 \end{aligned} \quad (3-12)$$

The value used for the shape factor g_i is 0.333.

Equation (2-23) contains contributions to the effective thermal conductivity from all solid constituents. In the simulator, these factors are replaced by one effective solid term with volume fraction $f = 1-\epsilon$. This is accomplished by defining the effective thermal conductivity of the solids $\bar{\lambda}$ and weighting factor \bar{k} by solving the equations

$$\bar{k}(1-\epsilon)\bar{\lambda} = \sum_i k_i f_i \lambda_i \quad (3-13)$$

$$\bar{k}(1-\epsilon) = \sum_i k_i f_i \quad (3-14)$$

Using previously stated values k_i , f_i , and λ_i , these equations can be solved, giving

$$\overline{k} = 0.329 \quad (3-15a)$$

$$\overline{\lambda} = 9.72 \times 10^{-3} \text{ (mcal/cm/sec/}^\circ\text{C)} \quad (3-15b)$$

The volumetric heat capacities of quartz and clay have the same value, $0.5 \text{ (cal/cm}^3\text{/}^\circ\text{C)}$, which was used for C_s in Equation (2-29) to compute the volumetric heat capacity of the soil-water system.

The soil moisture in the bottom layer was approximately constant over a 24-hour period; consequently, the option to hold this value constant in the simulation was chosen. (JBOT = 1 in NAMELIST, see Section 4.)

Values for parameters needed for the solution of the heat balance equation were either measured or could be deduced from the measurements. Air temperature, vapor pressure, and wind speed were measured at half-hour intervals, and a linear interpolation scheme was used to provide values at intermediate times. The subroutine which did this is listed in Section 4.

Values for parameters in the solar radiation flux model (Equation 2-37) are as follows: $\lambda = 33.5^\circ$, $\delta = 0.0^\circ$, $A = 0.3$, $N = 0.0$, and $n = 2.0$. The first two variables in this list follow from the location of the experiment (Phoenix) and the time of year (March 8, near the vernal equinox). Both incoming and outgoing short wave fluxes were measured, and the average value of their ratio (the albedo) over the day was 0.3. Figure 3-8 shows both modeled and measured fluxes. The qualitative agreement is very good. The major error source is the phase difference of one-half to one hour.

Evaporation rates were measured during the experiment. Therefore the constant C_1 in Equation 2-44 was chosen to make the model calculations match the data as accurately as possible when the rate was the largest. The simulated and measured evaporation rates are compared in Figure 3-9, with the value of $0.5 \times 10^{-6} \text{ (cal/cm}^3\text{/mb)}$ used for C_1 . In the nominal simulation C_0 was set equal to zero. The simulation was also run with the value of C_0 chosen to represent the reported

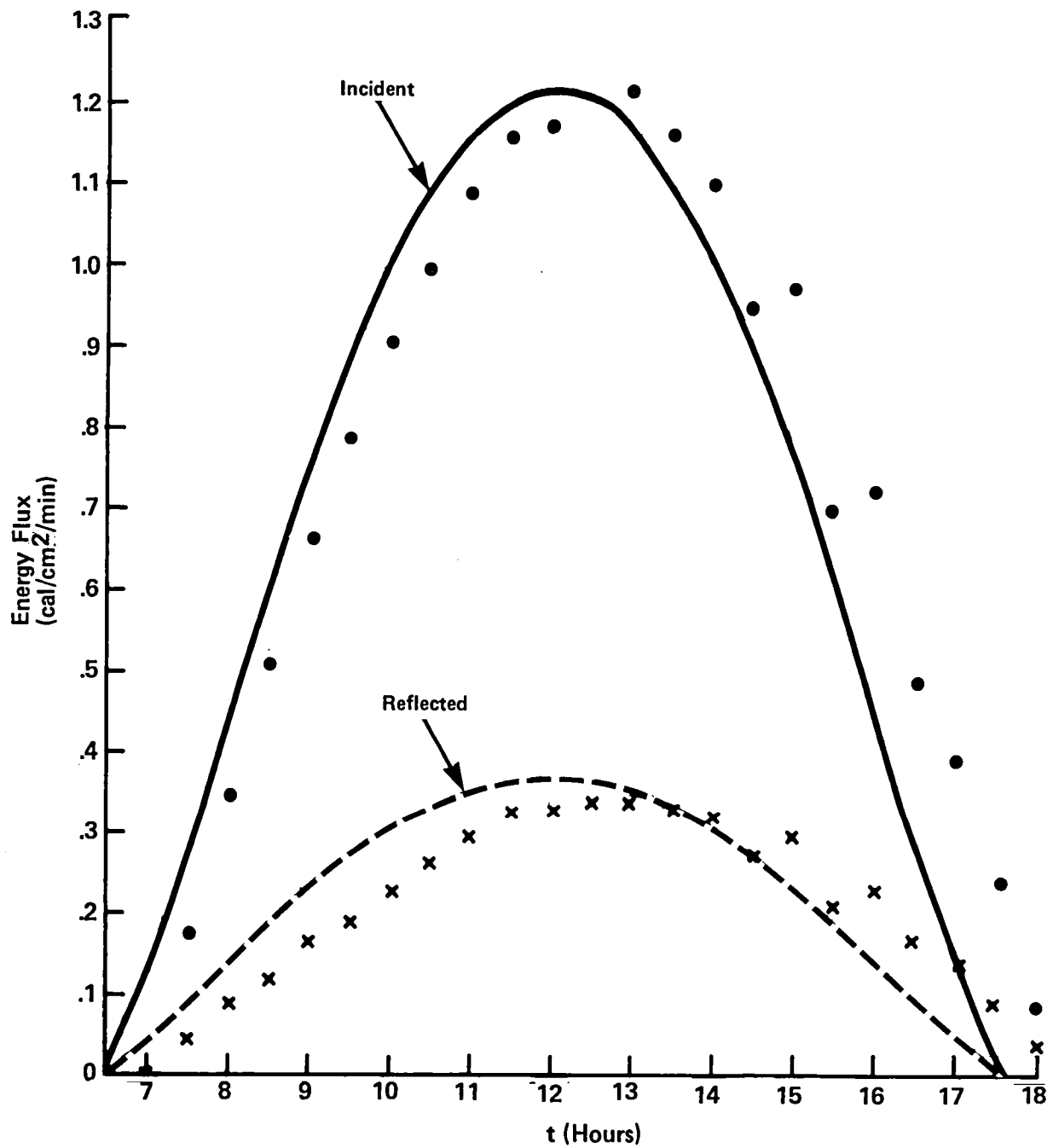


Figure 3-8. Simulated and measured incoming and outgoing short wave radiation fluxes for March 8 simulation. Solid lines represent the data, and the points represent the model calculations.

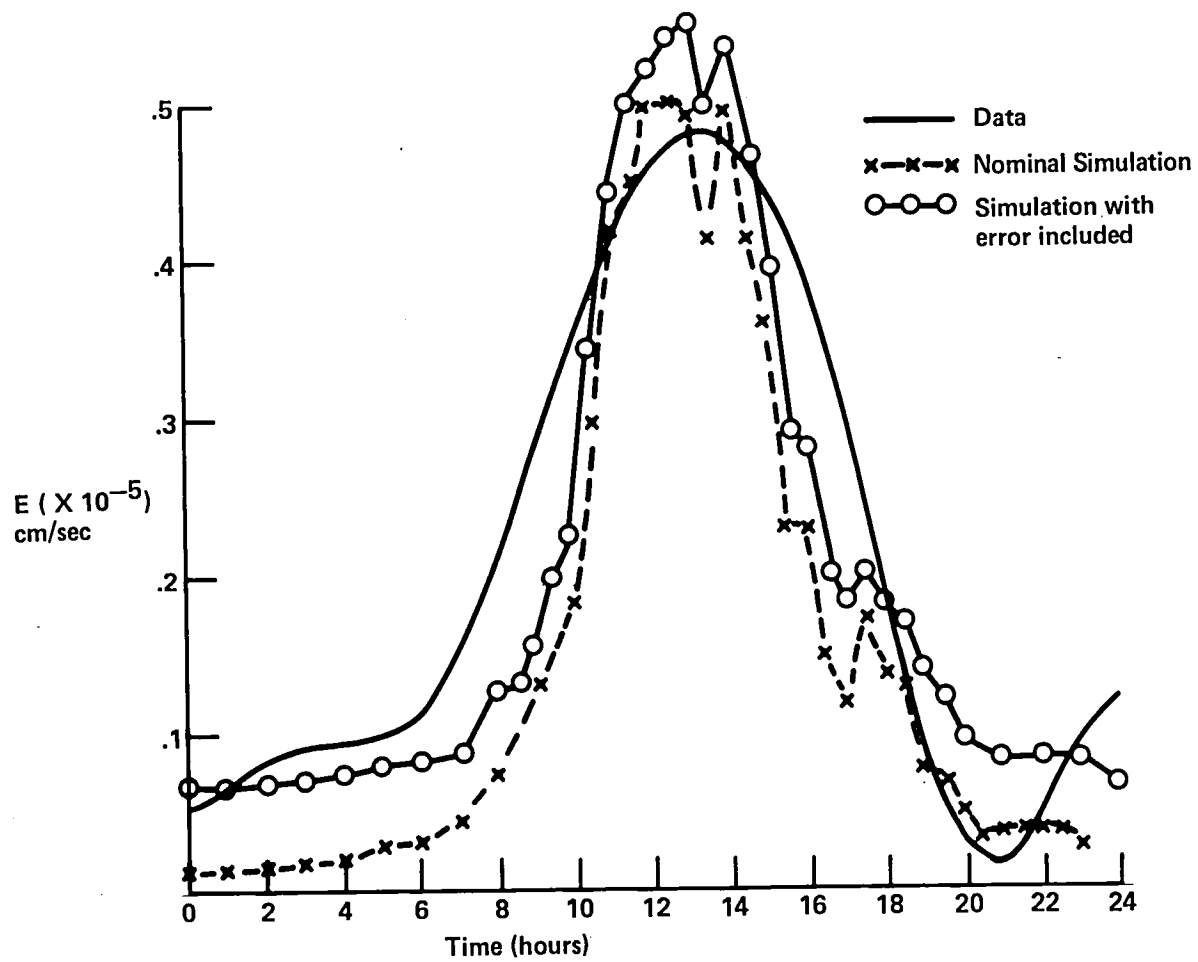


Figure 3-9. Comparison between simulated and measured evaporation rates.

uncertainty of .02 mm/hr., the resulting curve is also shown in Figure 3-9. The qualitative agreement is good, indicating that Equation 2-44 can be used to model physically realistic instantaneous evaporation rates.

The results of a 24-hour simulation are presented in Figures 3-10 through 3-15. The qualitative similarity between the measured and simulated variables is evidence that the program has solved the model equations correctly. As previously noted, the solutions of the model equations are not expected to agree exactly with the data because of uncertainties in soil properties.

Figure 3-10 shows simulated and measured wetness in the top $\frac{1}{2}$ cm of soil as a function of time. The greatest daily variation occurs in this layer, so a comparison of the simulated and measured wetness there provides the most stringent test of the program. Curve a is the data, and curve b is the simulation using nominal values of the input parameters. The quantitative agreement between these two curves is poor. However, some of the discrepancy can be accounted for by uncertainties in the input parameters. For example, the error in the evaporation rate is conservatively estimated to be .02 millimeters per hour (Jackson, 1972). Curve c results from increasing the magnitude of the evaporation rate in the simulation by this amount, and only a small change is produced. The values of the hydraulic parameters can also be in error by considerable amounts (Equation 3-10). Decreasing the magnitudes of K and ψ by the factors of .33 and .43 respectively (and therefore reducing the value of the wetness flux everywhere) results in curve d. The simulated wetness is very sensitive to changes in these variables. It should also be noted that the errors quoted include only those due to curve fitting and extracting numbers from the semilog plot; the errors in measuring $K(\theta)$ and $\psi(\theta)$ are not known and have therefore not been included. Finally, curve e shows the effect of changing both the evaporation rate and hydraulic parameters. In this case changing the evaporation rate has a greater effect on the moisture content.

Figure 3-11 shows the simulated and measured surface temperatures, and Figure 3-12 shows the corresponding heat fluxes at a depth of 5 cm. The qualitative agreement in both cases is good.

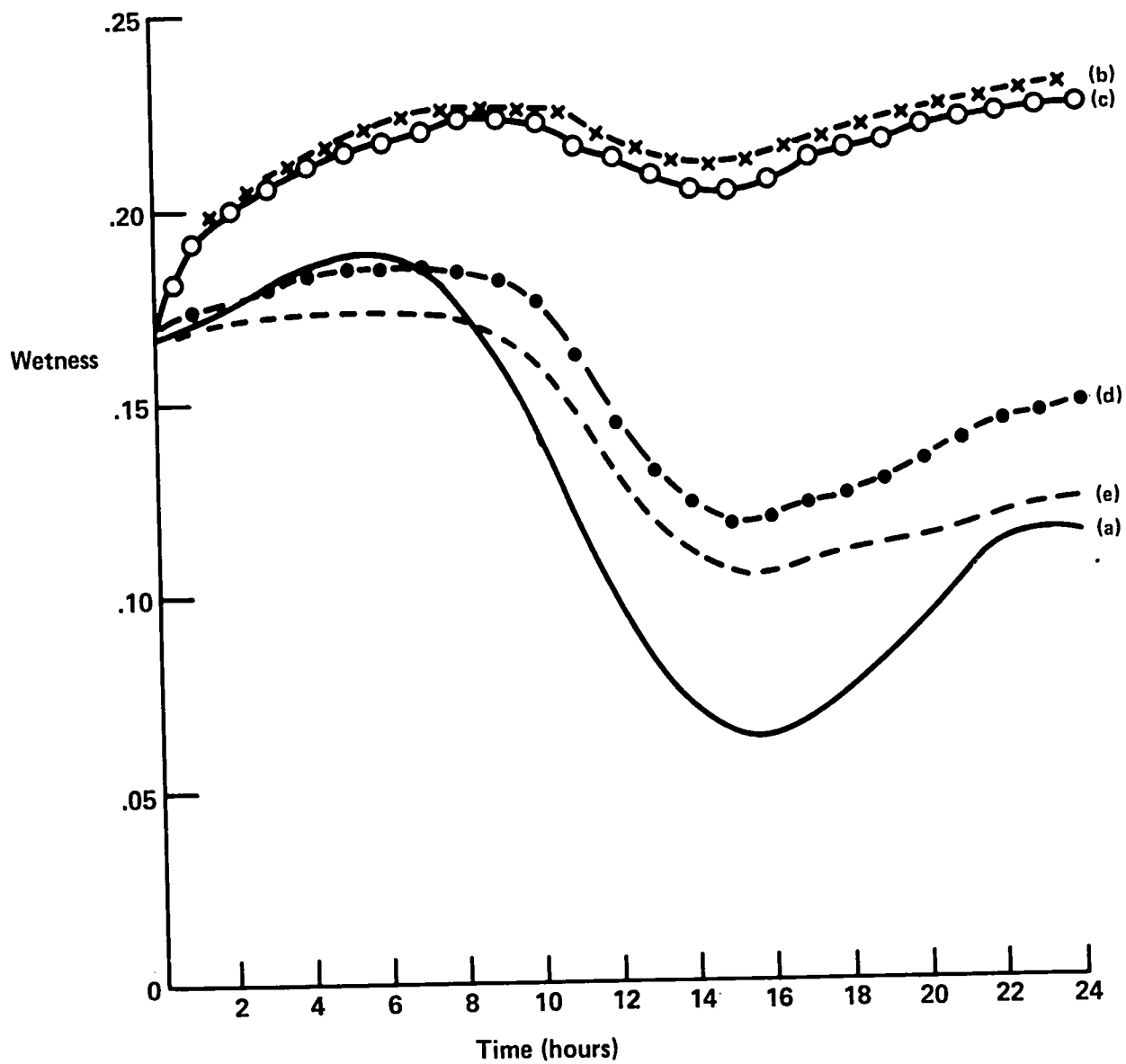


Figure 3-10. Wetness in top $\frac{1}{2}$ cm. of soil as a function of time. Curve a is the data, and curves b-e represent simulations with various uncertainties in the input variables included.

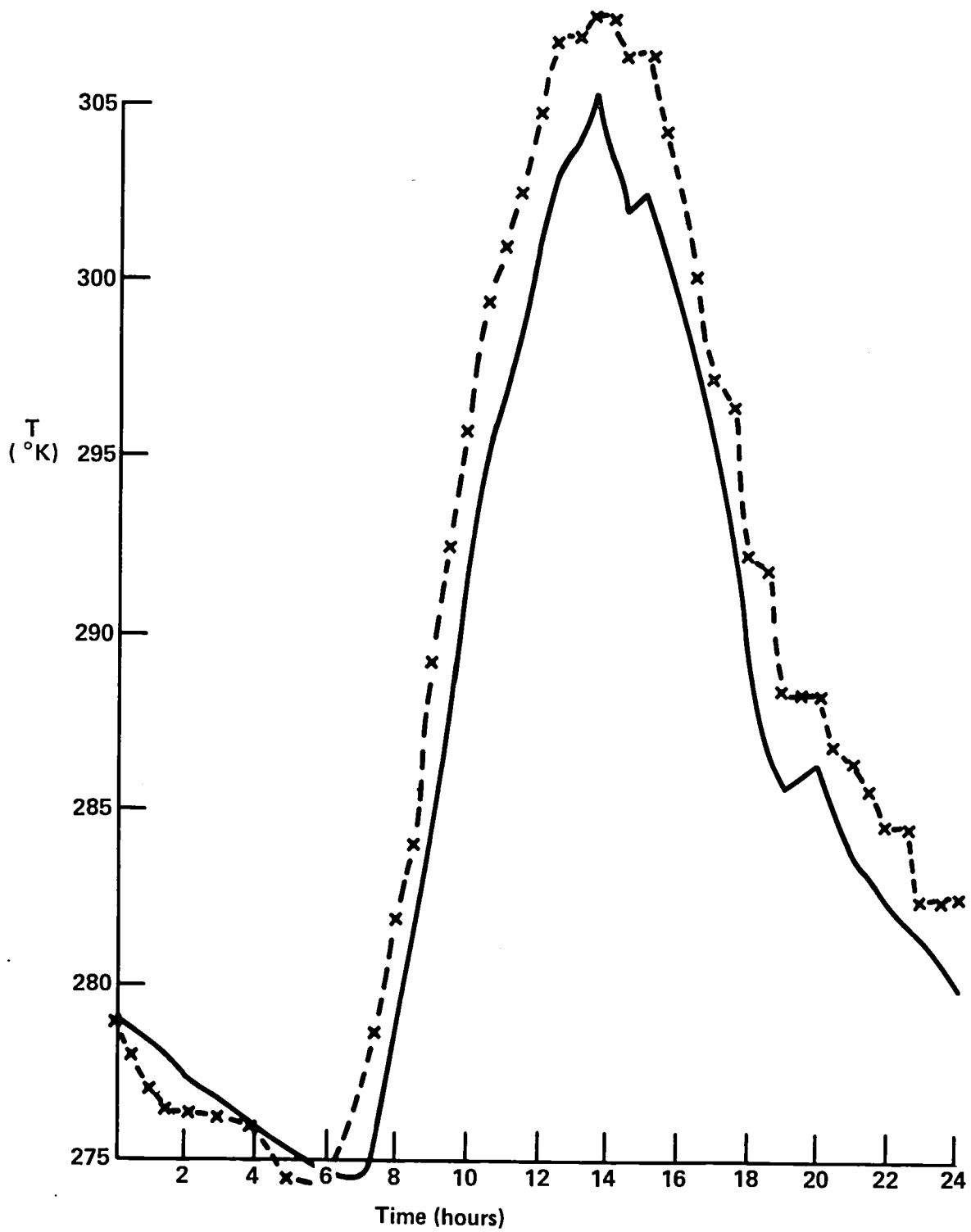


Figure 3-11. Simulated (dashed line) and measured (solid line) surface temperatures over one day.

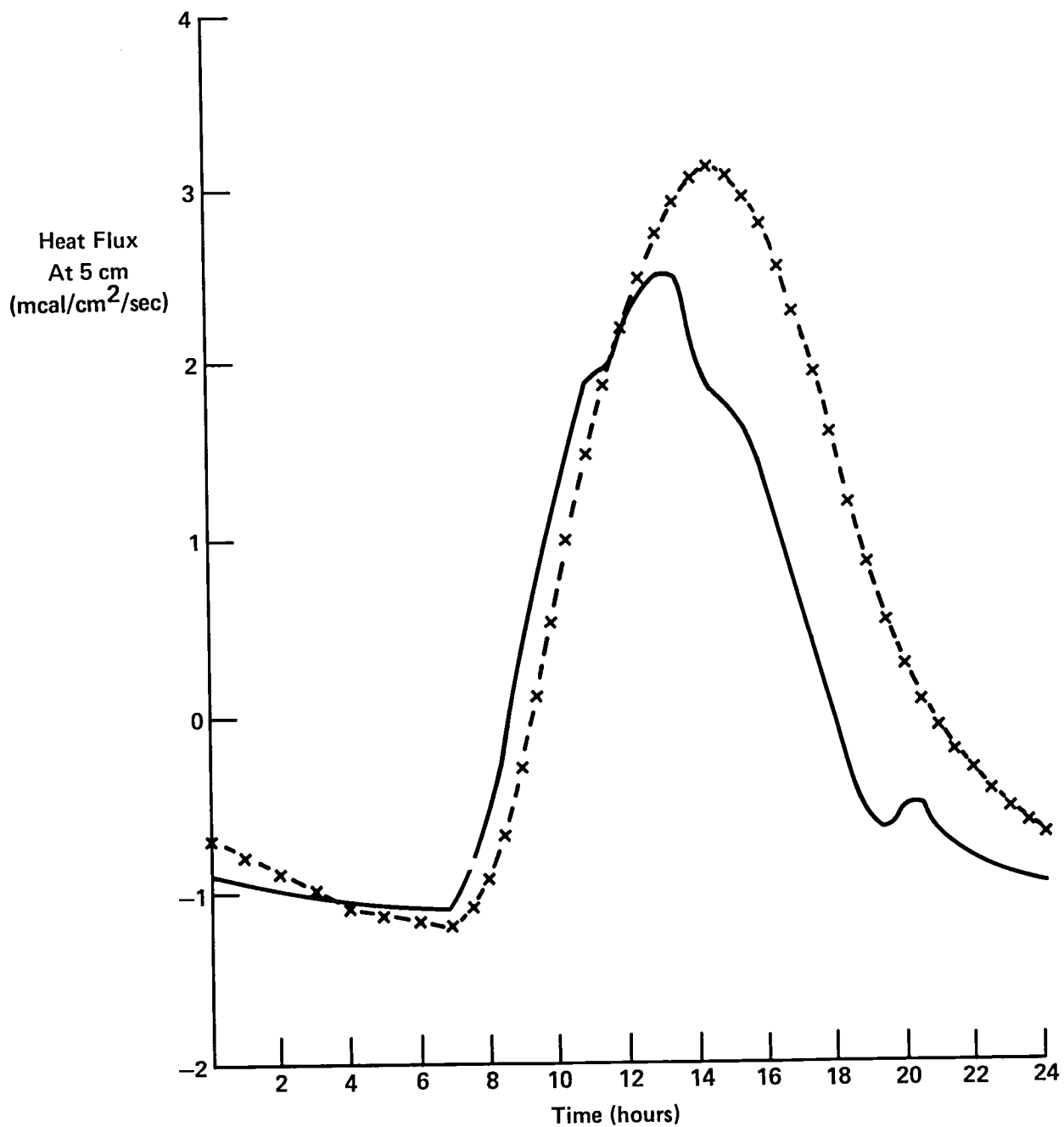


Figure 3-12. Simulated (dashed line) and measured (solid line) heat fluxes at 5 cm. depth. Fluxes are positive away from the surface.

The maximum error in the surface temperature is 3°C. Errors in the heat flux can be reduced by calibration of the thermal conductivity model (Equation 2-23). (Kimball, et al., 1976a and b.)

This is done either by making the air shape factors (Equation 2-24) functions of soil moisture, or by introducing a constant factor in Equation 2-23. However, in this simulation the value 1/3 was used for g_a for all values of wetness, and no multiplying factor was used.

Figures 3-13 and 3-14 show wetness profiles for three different times of the day. The simulations represented in figure 3-13 were performed with nominal values for the input parameters (curve b, figure 3-10); figure 3-14 shows the changes when errors in the hydraulic parameters are included (curve d, figure 3-10). Figure 3-15 shows temperature profiles from the nominal simulation. The simulations were performed to a depth of 100 cm but little variation was found below 25 cm for wetness and temperature in both the simulations and the data; therefore, profiles below this depth are not shown. The qualitative agreement is good for moisture profiles below 7 cm. The temperature profiles agree within 3° at all depths.

Uncertainties in the hydraulic parameters are responsible for most of the surface moisture difference between the simulations and the data; errors in the simulation of evaporation rate are of secondary importance. Surface soil moisture variations are determined by the difference in the evaporation rate and the recharge flux into the surface layer from below. Table 2-1 shows values for the integrals of these fluxes over one day for the four simulations discussed. The integrated evaporation rate from the data is also presented.

The differences between columns b and d show that changing the hydraulic parameters has very little effect on the evaporation rate but does alter the recharge flux. Therefore large variations in surface soil moisture can be created by changing the hydraulic parameters because the input to the layer changes, but output is relatively constant. (This decrease in recharge is reflected in the smaller change over time of the simulated moisture profiles in Figure 3-14 when compared to Figure 3-13). However, if only the evaporation rate is changed (column c), both surface and

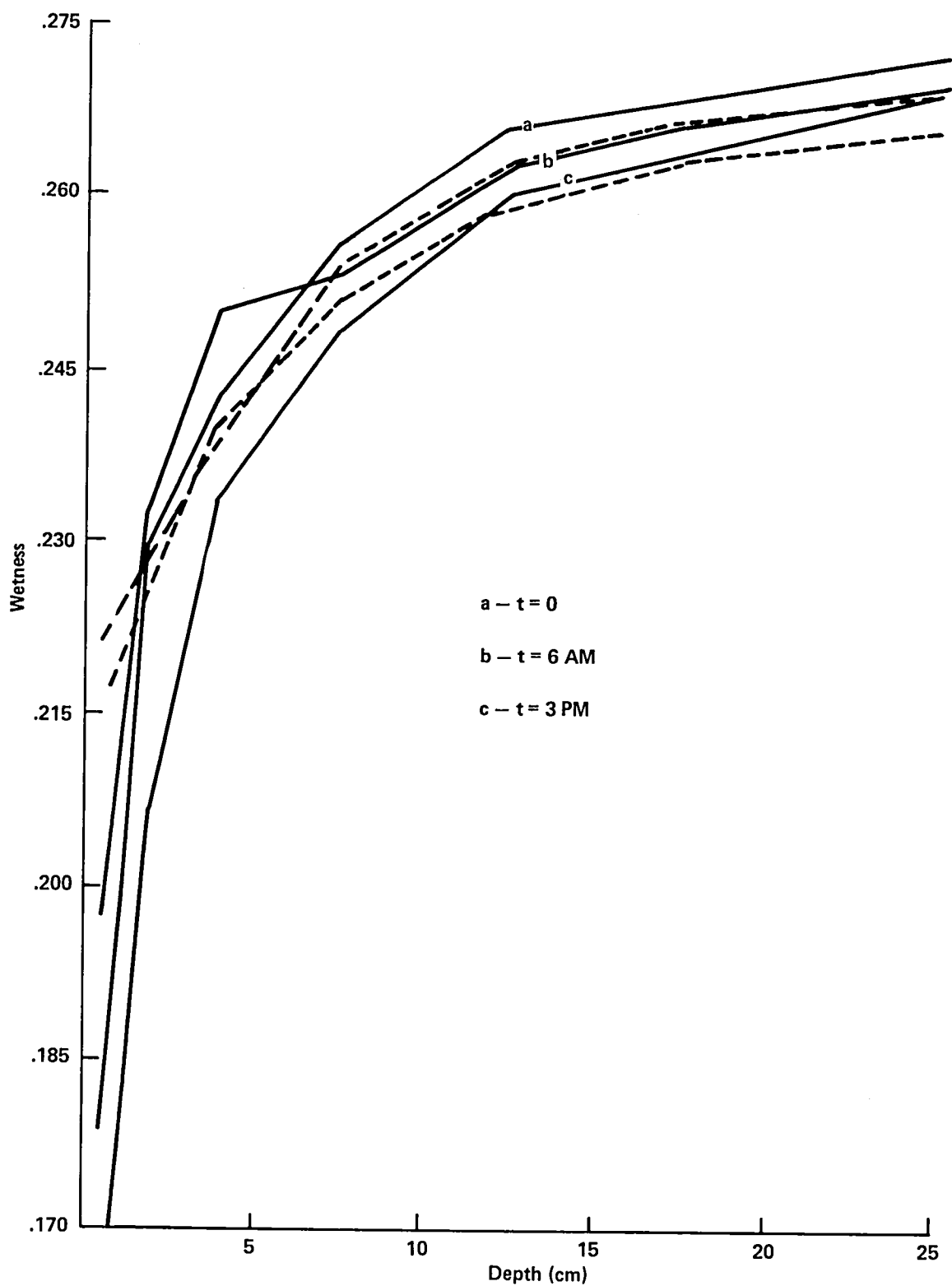


Figure 3-13. Wetness profiles at three times for the sample simulations. Solid lines are data, and dashed lines are simulations.

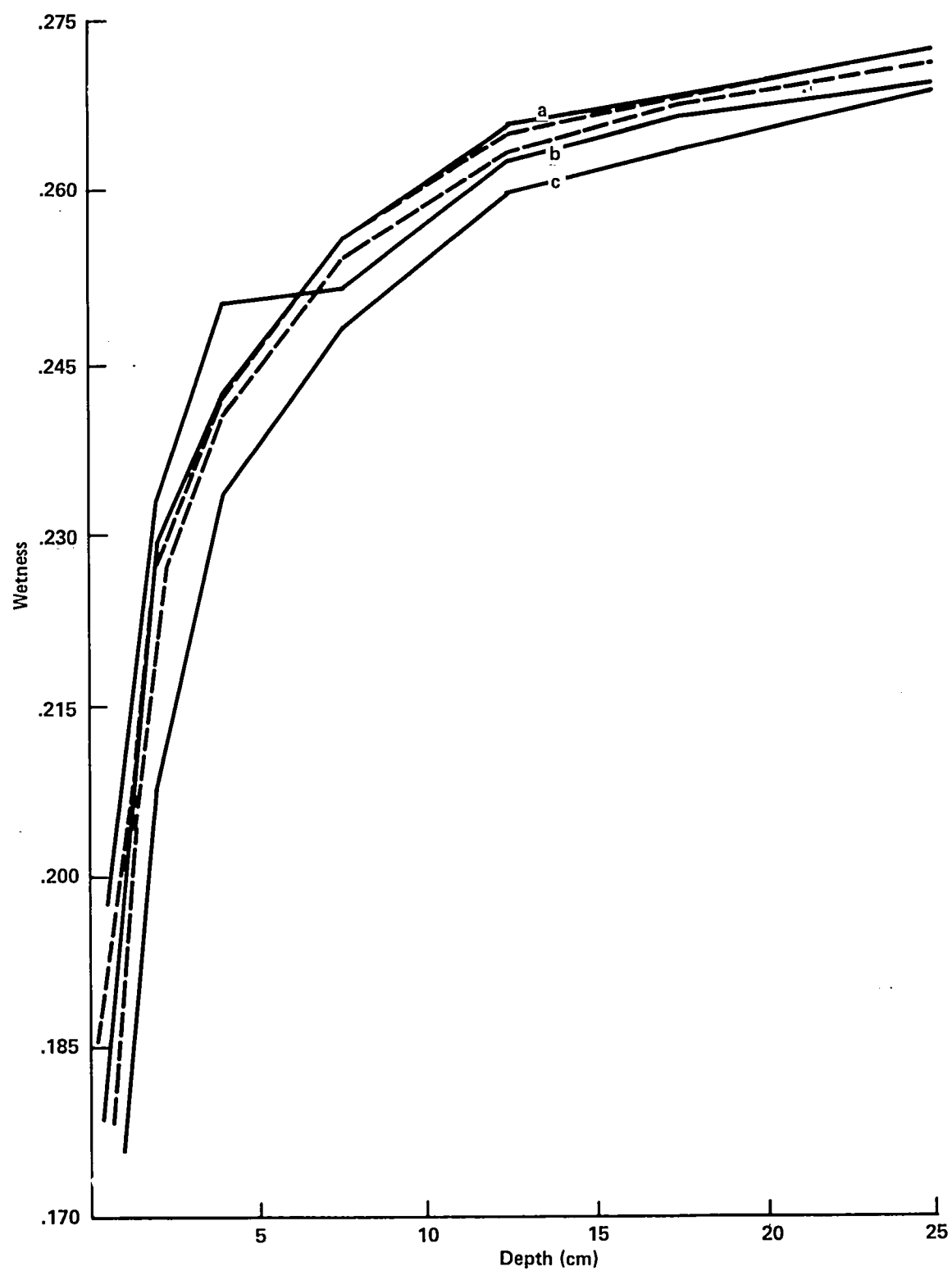


Figure 3-14. Wetness profiles at three times for a simulation with hydraulic parameter reduced by maximum uncertainty. The legend from figure 3-13 applies here.

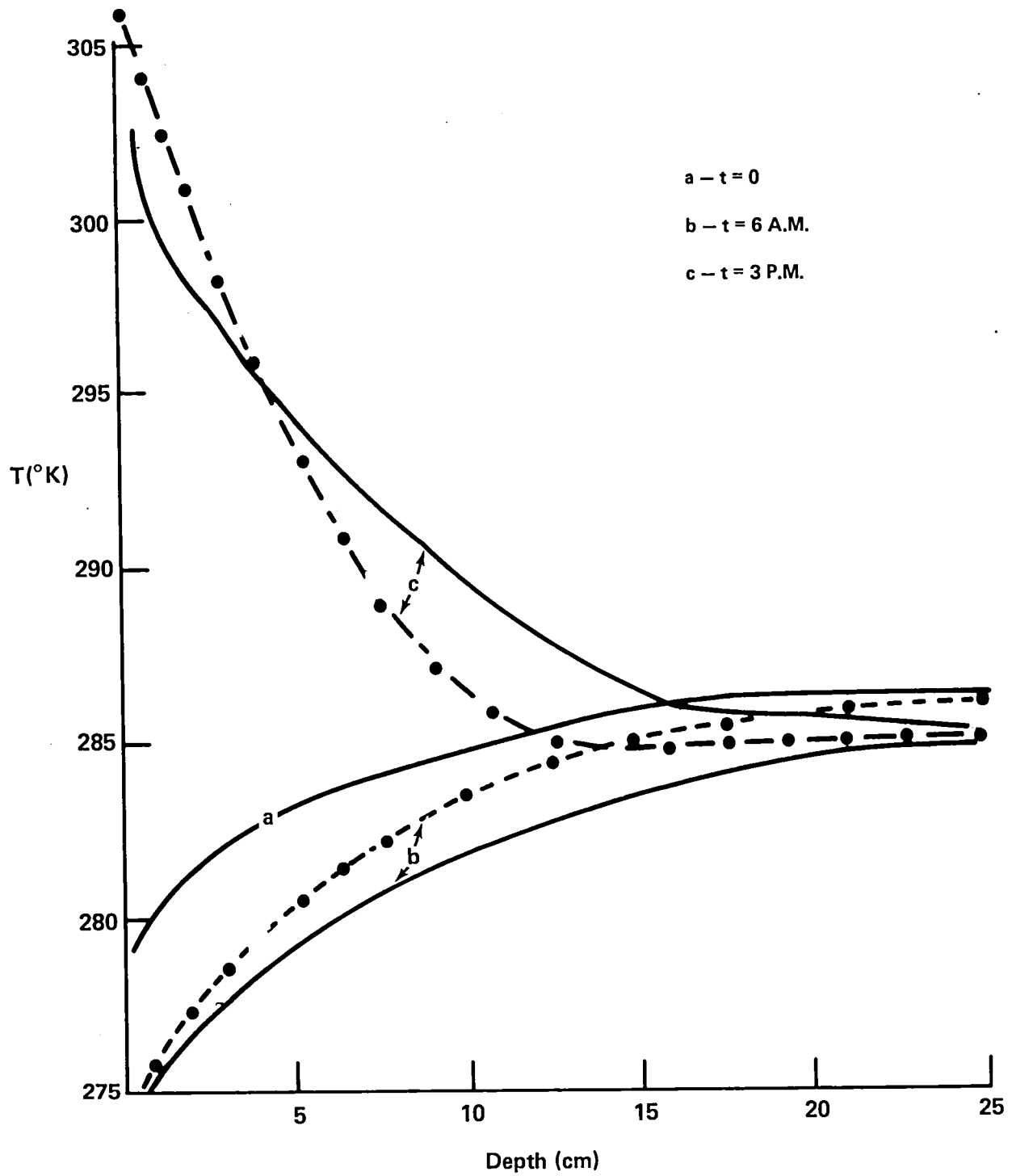


Figure 3-15. Temperature profiles at three times. Solid lines are data, and dashed lines are simulations.

Data Source

	a	b	c	d	e
Evaporation	.181	.125	.167	.126	.168
Recharge		.157	.196	.118	.147

Table 2-1. Daily Integrated Evaporation and surface recharge fluxes, in cm of water.

- a — Data
- b — Simulation with nominal input values
- c — Simulation with evaporation rate increased by uncertainty
- d — Simulation with hydraulic parameters reduced
- e — Simulation with evaporation rate increased and hydraulic parameters reduced

recharge fluxes change by almost the same amount. Thus the effect on surface soil moisture is minimal. The increased evaporation in this case is supplied by moisture in layers deeper than the surface.

SECTION 4 – USER'S GUIDE

Figure 4-1 shows the JCL needed to execute the simulator on the SACC (Science and Applications Computing Center, GSFC) S/360-91 and S/360-75 computers. Subroutines which perform the integration and create printer plots are in object module format as members of the data set ZB2PC.UTIL.OBJ. All other subroutines exist as a non-executable load module named ZB2PCSSM on the system load module data set, SYS2.LOADLIB. The linkage editor is used to combine these and create an executable load module. In this example NAMELIST input is entered instream in the GO.DATA5 DD statement. The DD card labeled GO.FT10F001 is for printer plots (see the description of NAMELIST parameter IPR). The GO.FT12F001 DD statement points to the disk data set which will receive the temperature and moisture profiles (see the description of NAMELIST parameters NDISK and IUDISK).

Figure 4-2 is an example of the JCL used to relink a subroutine created by the user and then execute the simulator. The first job step is a compilation of the user subroutine using the standard FORTRANH procedure, and the second step is the LINKGO procedure from Figure 4-1. Note that in the present example the NAMELIST is read from the data set ZB2PC.MARCH17.NL.DATA.

The variables in the simulator NAMELIST are described in Table 4-1. For each variable the type (real (R) or integer (I)) length (4 or 8 bytes), and default value are given. If the variable name and value does not appear in the NAMELIST input data, then the variable will have the value given in Table 4-1. Variable descriptions are also given. References to defining equations are provided where relevant.

In addition to values for the NAMELIST variables, the user can provide values for the air temperature, air vapor pressure, and wind speed as functions of time. This is done by coding a subroutine named METEOR. The routine which is used in the simulator is given in Figure 4-3. The calling sequence is as follows:

```

/* SOIL SIMULATE
//LGO EXEC LINKGO,REGION.GO=300K,OUT=8
//LINK.SYSLIB DD DISP=SHR,DSN=ZB2PC.UTIL.OBJ
//LINK.OBJECT DD *
INCLUDE LOADLIB(ZB2PCSSM)
ENTRY MAIN
/*
//GO.DATAS DD *
&INPUT
NL=22,DZ=1.0,2.0,3.0,4.0,18*5.0,WATER=27*0.1,TEMPS=27*290.0,
SATW=27*0.4,SATP=27*-10.0,SATK=27*1.0E-4,EB=27*5.0,
TSTOP=8.64D5,DTOUT=10800.0,
WATERR=0.002,TEMERR=2.0,ED=2.0D0,HMAX=1800.0D0,
IROOTS=0,ITEMPS=1,NWATRS=2,NWFLUX=2,NTEMPS=2,NTFLUX=2,
SPRES=1.0D6,ROOTS=27*0.1,
INDXW=1,2,INDXWF=1,2,INDXT=1,2,INDXTF=1,2,
NHCUMS=1,IXHCUM=1,NWCUMS=2,IXWCUM=1,2,
NDISK=1,IUDISK=12,
RNSTRT=8.64E4,RNSTOP=1.728E5,RNTOT=0.0,
NFUNCT=8,
&END
//GO.FT10F001 DD SYSOUT=8,DCB=(RECFM=VBA,LRECL=137,BLKSIZE=7265)
//GO.FT12F001 DD DISP=SHR,DSN=ZB2PC.PROFILE.NORAIN.DATA(DRYNORTS)
//

```

Figure 4.1. Execution JCL

```

//COMP EXEC FORTRANH,PARM=XREF,OUT=8
//SYSIN DD DISP=SHR,DSN=ZB2PC.METEOR.FORT
//LGO EXEC LINKGO,REGION.GO=300K,OUT=8
//LINK.SYSLIB DD DISP=SHR,DSN=ZB2PC.UTIL.OBJ
//LINK.OBJECT DD *
INCLUDE LOADLIB(ZB2PCSSM)
ENTRY MAIN
/*
//GO.DATAS DD DISP=SHR,DSN=ZB2PC.MARCH17.NL.DATA
//GO.FT10F001 DD SYSOUT=8,DCB=(RECFM=VBA,LRECL=137,BLKSIZE=7265)
//

```

Figure 4.2. Relink JCL

TABLE 4-1 NAMELIST INPUT VARIABLES

All subscripted variables are arrays with one element per soil layer. Subscripts run from 1 to NL (number of soil layers) unless otherwise indicated. The NAMELIST name is INPUT.

<u>Integrator Control Variables</u>			
<u>Name</u>	<u>Type</u>	<u>Default</u>	<u>Description</u>
TSTOP	R*8	8.64D4	Stop time for integration (seconds from start)
IFORCE	I*4	1	Force integration step size to remain less than HMAX (0 = no, 1 = yes)
HMAX	R*8	1.8D3	Maximum step size (seconds) (not applicable if IFORCE = 0)
H	R*8	1.0D0	Initial step size (seconds) (not applicable if IFORCE = 1; in this case H is automatically set equal to HMAX/512)
WATERR	R*4	1.0E-3	Error tolerance parameter for wetness in soil layers (Equation (2-59))
TEMERR	R*4	1.0	Error tolerance parameter for temperature in soil layers (Equation (2-59))
ED	R*8	5.0D0	Error window parameter (Equation (2-59))
IROOTS	I*4	0	Include water uptake by plant roots in the simulation (0 = no, 1 = yes)
ITEMPS	I*4	0	Temperature model indicator: 0 = no temperature in model, 1 = model soil temperature profile, 2 = use force-restore equations to model surface and average subsurface temperatures

<u>Name</u>	<u>Type</u>	<u>Default</u>	<u>Description</u>
JBOT	I*4	1	Bottom wetness boundary condition indicator: 0 = flux is zero, 1 = wetness is constant, 2 = flux is hydraulic conductivity of bottom layer

Output Control Variables

<u>Name</u>	<u>Type</u>	<u>Default</u>	<u>Description</u>
DTOUT	R*4	1800.0	Output period (seconds)
NFUNCT	I*4	0	Number of wetness and temperature profiles per plot page (0 ≤ NFUNCT ≤ 10)
NWATRS	I*4	0	Number of soil layers for which wetness is to be plotted as a function of time (0 ≤ NWATRS ≤ 10)
INDXW (I, I=1, 10)	I*4	10*0	Indices of soil layers for wetness versus time plots
NWFLUX	I*4	0	Number of soil boundaries for which wetness flux is to be plotted as a function of time (0 ≤ NWFLUX ≤ 10)
INDXWF (I, I = 1, 10)	I*4	10*0	Indices of soil boundaries for wetness flux versus time plots
NTEMPS	I*4	0	Number of soil layers for which temperature is to be plotted as a function of time (0 ≤ NTEMPS ≤ 10)
INDXT (I, I=1, 10)	I*4	10*0	Indices of soil layers for temperature versus time plots
NTFLUX	I*4	0	Number of soil boundaries for which heat flux is to be plotted as a function of time (0 ≤ NTFLUX ≤ 10)
INDXTF (I, I= 1, 10)	I*4	10*0	Indices of soil layers for heat flux versus time plots
ITABLE	I*4	1	Print tables of important variables (0 = no, 1 = yes)

<u>Name</u> IPR	<u>Type</u> I*4	<u>Default</u> 10	<u>Description</u> Output unit number for printer plots (tables are printed on unit 6)
WL ¹	R*4	0.0	Lower limit for wetness on plots (cm ³ /cm ³)
WH ¹	R*4	0.0	Upper limit for wetness on plots (cm ³ /cm ³)
WFL ¹	R*4	0.0	Lower limit for wetness flux on plots (cm/sec)
WFH ¹	R*4	0.0	Upper limit for wetness flux on plots (cm/sec)
TL ¹	R*4	0.0	Lower limit for temperature on plots (°K)
TH ¹	R*4	0.0	Upper limit for temperature on plots (°K)
TFL ¹	R*4	0.0	Lower limit for heat flux on plots (cal/cm ² /sec)
TFH ¹	R*4	0.0	Upper limit for heat flux on plots (cal/cm ² /sec)
NWCUMS	I*4	0	Number of layer boundaries for which cumulative wetness fluxes are to be computed (0 ≤ NWCUMS ≤ 10)
IXWCUM (I, I = 1, NWCUMS)	I*4	0	Indices of boundaries for which cumulative wetness is to be com- puted
NHCUMS	I*4	0	Number of layer boundaries for which cumulative heat fluxes are to be computed (0 ≤ NHCUMS ≤ 10)
IXHCUM (I, I = 1, NHCUMS)	I*4	0	Indices of boundaries for which cumulative heat flow is to be computed

¹ If the lower and upper limits for any variable are equal, the actual limits used are determined from the data being plotted.

<u>Name</u>	<u>Type</u>	<u>Default</u>	<u>Description</u>
NDISK	I*4	0	Flag to indicate if wetness and temperature profiles are to be written to disk or tape (0 = no, 1 = yes)
IUDISK	I*4	12	Unit number for output device (disk or tape)

Variables Defining Soil Profile and Properties

<u>Name</u>	<u>Type</u>	<u>Default</u>	<u>Description</u>
NL	I*4	200	Number of soil layers ($2 \leq NL \leq 200$)
DZ(I)	R*4	200*1.0	Thickness of soil layers (cm)
WATER(I)	R*4	200*0.25	Initial volumetric wetness of soil layers (cm^3/cm^3)
TEMPS(I)	R*4	200*293.0	Initial temperature of soil layers ($^{\circ}\text{K}$)
SATW(I)	R*4	200*0.3	Saturation volumetric wetness of soil layers (cm^3/cm^3) (θ_s , Equation 2-65)
SATK(I)	R*4	200*1.0E-4	Saturation hydraulic conductivity of soil layers (cm/sec) (k_s , Equation (2-65a))
SATP(I)	R*4	200*-10.0	Saturation matric potential of soil layers (cm) (ψ_s , Equation (2-65b))
EB(I)	R*4	200*5.0	Texture parameter of soil layers b, (Equation (2-65))
PORSTY(I)	R*4	200*0.45	Porosity of soil layers (cm^3/cm^3)
TCONDS(I)	R*4	200*2.5E-3	Thermal conductivity of solid matter in soil layers ($\text{cal}/\text{cm}/\text{sec}/^{\circ}\text{K}$)
VHCAPS(I)	R*4	200*0.5	Volumetric heat capacity of soil layers ($\text{cal}/\text{cm}^3/^{\circ}\text{K}$)
FACTKS	R*4	0.75	Shape factor for soil grains (k_i in Equation (2-23))

<u>Name</u>	<u>Type</u>	<u>Default</u>	<u>Description</u>
TCONDW	R*4	1.3E-3	Thermal conductivity of water (cal/cm/sec/°K)
TCONDA	R*4	5.967E-5	Thermal conductivity of air (cal/cm/sec/°K)
VHCAPW	R*4	1.0	Volumetric heat capacity of water (cal/cm ³ /°K)
VHCAPA	R*4	3.0E-4	Volumetric heat capacity of air (cal/cm ³ /°K)
FACTKA	R*4	1.4	Air weighting factor in definition of thermal conductivity (k_a in Equation (2-28))
ALPHA	R*4	0.667	Tortuosity factor (α , Equation (2-6))
GAMMA0	R*8	2.09D-3	Surface tension temperature coefficient (γ , Equation (2-11))
RHOVP0	R*4	6.0035	Constant in exponential to compute the density of water vapor (Equation (2-7b))
RHOVPT	R*4	4975.9	Coefficient of 1/T in definition of the density of water vapor (Equation (2-7b))
DATM0	R*4	0.229	Diffusion coefficient of water vapor in air at 0° C (cm ² /sec) (Equation (2-6b))
LHEAT	R*8	586.0D0	Latent heat of vaporization of water (cal/gm)
THMIN	R*8	0.05D0	Minimum value of soil moisture to support evaporation. If the surface soil moisture is below this value, evaporation is limited to flux into the surface layer from below
SFRAC	R*8	0.1D0	Fraction of evapotranspiration demand which is satisfied by evaporation (f , Equation (2-63))
.ROOTS(I)	R*8	200*0.0D0	Root density profile (P_j , Equation (2-60))

<u>Name</u>	<u>Type</u>	<u>Default</u>	<u>Description</u>
CPMIN	R*8	-1.5D4	Limiting value of crown potential, cm (ϕ_w , Equation (2-62))
SPRES	R*8	1.0D6	Specific resistance of roots, (sec/cm) (r, Equation (2-58))

Variables Defining the Processes at the Air/Soil Interface

<u>Name</u>	<u>Type</u>	<u>Default</u>	<u>Description</u>
EMAX	R*8	3.0D-5	Maximum rate for time dependent evapotranspiration model (cm/sec) (E_{max} , Equation (2-50))
EMAXT	R*8	4.68D4	Time of maximum rate, seconds since start of simulation (t_{max} , Equation (2-50))
EDAY	R*4	1.0	Total daily evapotranspiration (cm) (E_{day} , Equation (2-51))
NRAINS	I*4	0	Number of rain storms
RNSTRT (10)	R*4	0.0	Start time of rainfall, seconds since start of simulation (t_o , Equation (2-49))
RNSTOP (10)	R*4	0.0	Stop time of rainfall, seconds since start of simulation (t_1 , Equation (2-49))
RNTOT (10)	R*4	0.0	Total rainfall accumulation, (cm) (r_{tot} , Equation (2-49))
ATTEN (10)	R*4	0.5	Short wave attenuation during rainfall
CEVAPO	R*4	0.0	Constant term in evapotranspiration model (cal/cm ² /sec) (C_o , Equation (2-44))
CEVAPI	R*4	0.0	Coefficient of variable term in evapotranspiration model (C_1 , Equation (2-44))
XLAT	R*4	45.0	Latitude of simulation, degrees (ϕ , Equation (2-39))
SUNDEC	R*4	0.0	Declination of Sun, degrees (δ , Equation (2-39))

<u>Name</u>	<u>Type</u>	<u>Default</u>	<u>Description</u>
ALB	R*4	0.3	Surface albedo to shortwave radiation (A, Equation (2-37))
CTRANS	R*4	0.2	Cloud transmittivity (k, Equation (2-37))
CLOUDS	R*4	0.0	Fractional cloud cover (N, Equation (2-37))
TURB	R*4	2.0	Air turbidity factor, (n, Equation (2-38))

```

SUBROUTINE METEOR(T,TAIR,VAPZO,WIND)
REAL*8 T
DIMENSION TEMPS(50),VAPORS(50),WINDS(50)
DATA TEMPS/
* 0.28036E 03, 0.28056E 03, 0.28026E 03, 0.28046E 03, 0.27976E 03,
* 0.27846E 03, 0.28166E 03, 0.27586E 03, 0.27806E 03, 0.27746E 03,
* 0.27796E 03, 0.27896E 03, 0.27866E 03, 0.27856E 03, 0.27806E 03,
* 0.28106E 03, 0.28306E 03, 0.29236E 03, 0.30076E 03, 0.29846E 03,
* 0.29806E 03, 0.30126E 03, 0.30206E 03, 0.30196E 03, 0.30396E 03,
* 0.30566E 03, 0.30576E 03, 0.30546E 03, 0.30296E 03, 0.30096E 03,
* 0.30126E 03, 0.30186E 03, 0.30266E 03, 0.30216E 03, 0.30296E 03,
* 0.30146E 03, 0.29936E 03, 0.29606E 03, 0.28896E 03, 0.28786E 03,
* 0.28876E 03, 0.28666E 03, 0.28556E 03, 0.28706E 03, 0.28796E 03,
* 0.28656E 03, 0.28736E 03, 0.28206E 03, 283.26E0, 283.06E0/
DATA VAPORS/
* 0.49000E 01, 0.44000E 01, 0.32000E 01, 0.96000E 01, 0.57000E 01,
* 0.25000E 01, 0.63000E 01, 0.49000E 01, 0.50000E 01, 0.55000E 01,
* 0.53000E 01, 0.55000E 01, 0.52000E 01, 0.55000E 01, 0.55000E 01,
* 0.77000E 01, 0.61000E 01, 0.67000E 01, 0.76000E 01, 0.13500E 02,
* 0.53000E 01, 0.59000E 01, 0.56000E 01, 0.57000E 01, 0.55000E 01,
* 0.53000E 01, 0.57000E 01, 0.55000E 01, 0.57000E 01, 0.57000E 01,
* 0.52000E 01, 0.51000E 01, 0.50000E 01, 0.48000E 01, 0.47000E 01,
* 0.46000E 01, 0.44000E 01, 0.44000E 01, 0.59000E 01, 0.59000E 01,
* 0.49000E 01, 0.47000E 01, 0.51000E 01, 0.46000E 01, 0.43000E 01,
* 0.45000E 01, 0.45000E 01, 0.51000E 01, 4.8E0,4.4E0/
DATA WINDS/
* 59.0, 68.6, 85.4, 94.8, 104.8, 115.8, 136.4, 114.8, 98.8,
* 112.4, 93.0, 60.2, 91.4, 98.2, 74.6, 88.0, 100.0, 79.2,
* 99.0, 125.0, 107.2, 104.8, 112.6, 88.4, 54.2, 60.4, 50.6,
* 40.6, 67.2, 90.4, 112.8, 139.8, 162.4, 146.8, 127.4, 102.0,
* 93.4, 77.0, 77.2, 83.8, 89.2, 78.0, 75.4, 68.0, 62.2,
* 58.6, 58.6, 60.8, 59.2, 64.0/
RINTRP(A,B) = A + (B - A)*DTREL
TREL = DMOD(T,8.64D4)/1800.0D0
ITRL = INT(TREL)
IO = ITRL + 1
I1 = IO + 1
DTREL = TREL - ITRL
TAIR = RINTRP(TEMPS(IO),TEMPS(I1))
VAPZO = RINTRP(VAPORS(IO),VAPORS(I1))
WIND = RINTRP(WINDS(IO),WINDS(I1))
RETURN
END

```

Figure 4-3. Example of subroutine METEOR

T – input – Time in seconds since start of simulation (R*8).
 TAIR – output – Air temperature at reference height above the surface, degrees K, (R*4).
 VAPZO – output – Air vapor pressure at reference height, millibars (R*4).
 WIND – output – Wind speed at reference height, cm/sec (R*4)

The output variables are used to solve the heat balance equation. In this example data are provided at half-hour intervals, and a linear interpolation scheme is used to provide values at other times. The DMOD function causes the simulator to reuse the data in case the simulation extends beyond 24 hours.

As long as the calling sequence convention is followed and the data have the correct units, the user is free to use any scheme to generate values. The relink JCL of Figure 4-2 would be used to link the user subroutine with the simulator. The subroutine would be on the sequential data referred to by the SYSIN DD statement of the FORTRANH procedure.

Figures 4-4 through 4-12 are samples of the printed output created by the simulator. Figure 4-4a shows values of input parameters which vary with depth. In the following, NAMELIST names are given in parentheses. The first three columns are the layer index, the layer thickness (DZ), and the depth to the center of each layer. The latter is computed by the simulator using the input layer thicknesses. The next two columns are the initial wetness (WATERS) and the temperature (TEMPS) of each layer. The next three columns are the thermal conductivity of the solid material (TCONDS), porosity (PORSTY), and the volumetric heat capacity of the solids (VHCAPS). The next four columns are used in the model of the soil hydraulic properties, Equations 2-65a and b. They are saturation wetness θ_s (SATW), saturation matric potential ψ_s (SATP), saturation hydraulic conductivity K_s (SATK), and soil texture parameter b (EB). The column labeled EM is the exponent $2b + 3$ in the hydraulic conductivity model. The last column (ROOTS) is the root density profile, P_i from Equation 2-60.

** INPUT AND INITIALIZED PARAMETERS **

I	THICKNESS	DEPTH	WETNESS	TEMP	TCONDS	PORSTY	VHCAP	SATW	SATP	SATK	ER	EM	ROOTS									
10.500E	00	0.2500	00	0.167E	00	0.279E	03	0.250E-02	0.450E	03	0.500E	00	0.375E	00	-1.142E	02	0.122E-04	0.500E	01	0.130E	02	0.0
20.500E	00	0.7500	00	0.193E	00	0.280E	03	0.250E-02	0.450E	03	0.500E	00	0.375E	00	-1.142E	02	0.122E-04	0.500E	01	0.130E	02	0.0
30.500E	01	0.2000	01	0.229E	00	0.281E	03	0.250E-02	0.450E	03	0.500E	00	0.375E	00	-1.142E	02	0.122E-04	0.500E	01	0.130E	02	0.0
40.500E	01	0.4000	01	0.243E	00	0.283E	03	0.250E-02	0.450E	03	0.500E	00	0.375E	00	-1.142E	02	0.122E-04	0.500E	01	0.130E	02	0.0
50.500E	01	0.7500	01	0.256E	00	0.284E	03	0.250E-02	0.450E	03	0.500E	00	0.375E	00	-1.142E	02	0.122E-04	0.500E	01	0.130E	02	0.0
60.500E	01	0.1250	02	0.266E	00	0.285E	03	0.250E-02	0.450E	03	0.500E	00	0.375E	00	-1.142E	02	0.122E-04	0.500E	01	0.130E	02	0.0
70.500E	01	0.1750	02	0.268E	00	0.286E	03	0.250E-02	0.450E	03	0.500E	00	0.375E	00	-1.142E	02	0.122E-04	0.500E	01	0.130E	02	0.0
80.100E	02	0.2500	02	0.272E	00	0.286E	03	0.250E-02	0.450E	03	0.500E	00	0.375E	00	-1.142E	02	0.122E-04	0.500E	01	0.130E	02	0.0
90.100E	02	0.3500	02	0.265E	00	0.286E	03	0.250E-02	0.450E	03	0.500E	00	0.375E	00	-1.142E	02	0.122E-04	0.500E	01	0.130E	02	0.0
100.100E	02	0.4500	02	0.258E	00	0.286E	03	0.250E-02	0.450E	03	0.500E	00	0.375E	00	-1.142E	02	0.122E-04	0.500E	01	0.130E	02	0.0
110.100E	02	0.5500	02	0.244E	00	0.287E	03	0.250E-02	0.450E	03	0.500E	00	0.375E	00	-1.142E	02	0.122E-04	0.500E	01	0.130E	02	0.0
120.100E	02	0.6500	02	0.239E	00	0.287E	03	0.250E-02	0.450E	03	0.500E	00	0.375E	00	-1.142E	02	0.122E-04	0.500E	01	0.130E	02	0.0
130.100E	02	0.7500	02	0.214E	00	0.287E	03	0.250E-02	0.450E	03	0.500E	00	0.375E	00	-1.142E	02	0.122E-04	0.500E	01	0.130E	02	0.0
140.100E	02	0.8500	02	0.170E	00	0.288E	03	0.250E-02	0.450E	03	0.500E	00	0.375E	00	-1.142E	02	0.122E-04	0.500E	01	0.130E	02	0.0
150.100E	02	0.9500	02	0.153E	00	0.288E	03	0.250E-02	0.450E	03	0.500E	00	0.375E	00	-1.142E	02	0.122E-04	0.500E	01	0.130E	02	0.0

Figure 4-4a. Input Parameters

Figure 4-4b is an example of the second output page, where the values of all other input parameters are given. The variable name appears first, followed by a brief description and its value.

A typical page of table output is shown in Figure 4-5. The output time is in DDDHHMMSS.SS format, where DDD is the day (DDD = 0 is the first day), HH is the hour of the day, MM is minutes of the hour, and SS.SS is seconds. It is assumed that the simulation start time corresponds to midnight. The first line also shows the current integrator step size in seconds.

Next on this table are the values of some important variables related to moisture flow. Layer number, depth to the center of each layer, and wetness in the layer are shown in the first three columns. Next is the wetness flux in cm/sec (q_0 Equation 2-75a) at each of the NL + 1 layer boundaries. The first layer is the air-soil interface. The next two columns are the hydraulic conductivity (HYD COND) and matric potential, or pressure head (PHEAD) of the water in the layer. The next column lists values of the derivative of the matric potential with respect to depth ($\frac{d\psi}{dz}$ Equation 2-73a). The I^{th} derivative is evaluated at the boundary separating the I and I + 1 layers. The last entry in the column is not used, since the derivative is required only at the NL-1 interior boundaries. The column labeled DWDT is the derivative of wetness with respect to time ($\frac{d\theta}{dt}$, Equation 2-54). If the sink term is zero, this is the spatial derivative of the wetness flux ($-\frac{dq}{dz}$, Equation 2-72a). The last column is the value of the sink term Q, Equation 2-55.

The next entry is the amount of water stored in the profile in centimeters. This will be calculated if the value of the input variable NWCUMS is greater than zero.

The cumulative wetness flux at any of up to 10 layer boundaries can be computed. This is the integral of the flux over time, starting with the beginning of the simulation. The trapezoid rule is used to compute the integral, and the value of the integral is updated after each simulator time step. The number of integrated fluxes is set via NAMELIST parameter NWCUMS; the layer indices are input via the array IXWCUM. In the example of Figure 4-5 NWCUMS is 3, and the first three elements of IXWCUM are 1, 2, and 16 (see Figure 4-4b). Boundary 1 is the air-soil interface, so

```

INTEGRATOR VARIABLES
TSTOP  STOP TIME (SECONDS) 0.864000D 05
WATERR ERROR TOLERANCE FOR WETNESS 0.20E-02
TEMERR  ERROR TOLERANCE FOR TEMPERATURE C.20E 01
ED      ERROR WINDOW: WINDOW = (E/ED,E*ED) 0.20D 01
IFORCE  FORCE MAX. STEP SIZE (0=NO, 1=YES) 1
H       INITIAL STEP SIZE (N/A IF IFORCE=1) 0.100000D 01
HMAX    MAXIMUM STEP SIZE (N/A IF IFORCE=0) 0.180000D 04

OUTPUT CONTROL VARIABLES
DTOUT  OUTPUT PERIOD (SECONDS) 0.180000E 04
NFUNCT NUMBER OF GRAPHS OF WETNESS VS DEPTH AND TEMP VS DEPTH PER PLOT PAGE 6
NWAtrs NUMBER OF WETNESS VS TIME GRAPHS 3
INDXW  SOIL LAYERS FOR WETNESS PLOT 1 2 3 0 0 0 0 0 0 0
NWFLUX NUMBER OF WETNESS FLUX VS TIME GRAPHS 2 1 2 0 0 0 0 0 0 0
INDXWF BOUNDARIES FOR WETNESS FLUX PLOT 1 2 0 0 0 0 0 0 0 0
NTEMPs NUMBER OF TEMP VS TIME GRAPHS 3 1 2 3 0 0 0 0 0 0
INDXT  SOIL LAYERS FORT TEMP PLOT 1 2 3 0 0 0 0 0 0 0
NTFLUX NUMBER OF HEAT FLUX VS TIME GRAPHS 3 1 2 5 0 0 0 0 0 0
INDXTF BOUNDARIES FORT HEAT FLUX PLOT 1 2 5 0 0 0 0 0 0 0
IPR     OUTPUT UNIT NUMBER FOR ALL PLOTS 10
ITABLE  PRINT TABLES (0=NO,1=YES) 1
WL,WH  FORCED MIN AND MAX FOR WETNESS PLOT 0.0 0.0
TL,TH  FORCED MIN AND MAX FOR TEMP. PLOT 0.0 0.0
WFL,WFH FORCED MIN AND MAX FOR W FLUX PLOT 0.0 0.0
TFL,TFH FORCED MIN AND MAX FOR H FLUX PLOT 0.0 0.0
IF L AND H LIMITS ARE EQUAL, THEY ARE SET BY THE PROGRAM

NWCUMS NUMBER OF CUMULATIVE WETNESS FLUXES 3
IXWCUM BOUNDARIES FOR CUMULATIVE WETNESS FLUXES 1 2 16 0 0 0 0 0 0 0
NHCUMS NUMBER OF CUMULATIVE HEAT FLUXES 1
IXHCUM BOUNDARIES FOR CUMULATIVE HEAT FLUXES 1 0 0 0 0 0 0 0 0 0
NDISK  OUTPUT STATE TO DISK (0=NO,1=YES) 0
IUDISK FORTRAN UNIT NUMBER FOR DISK OUTPUT 12

OTHER MODEL PARAMETERS
ITEMPS TEMPERATURE MODEL INDICATOR (0=NO TEMP., 1=TEMP.PROFILE, 2=FORCE-RESTORE) 1
IROOTS INCLUDE ROOT MODEL (0=NO, 1=YES) 0
SPRES  ROOT SPECIFIC RESISTANCE (SEC/CM) 0.100000D 07
CPMIN  PLANT CROWN POTENTIAL AT WILTING -0.150000D 05
SFRAC  FRACTION OF ET DEMAND TO SOIL EVAP 0.100000D 00
NEXT 3 PARAMETERS USED FOR GAUSSIAN TIME-DEPENDENT ET FUNCTION WHEN ITEMPS = 0
EMAX  MAXIMUM RATE (CM/SEC) 0.300000E-04
EDAY  TOTAL DAILY EVAPOTRANSPIRATION (CM) 0.100000E 01
EMAXT TIME OF MAX RATE (SEC SINCE START) 0.468000E 05
THMIN LIMITING SURFACE WETNESS FOR EVAPORATION 0.500000D-01
RNTOT  TOTAL RAIN FALL (CM) 0.0
RNSTRT START TIME OF RAIN (SEC FROM SIM START) 0.0
RNSTOP  STOP TIME OF RAIN (SEC FROM SIM START) 0.0
RNRATE RAINFALL RATE (CM/SEC) 0.0
JBOT  BOTTOM WETNESS BOUND. COND. 1
=0,FLUX=0.0; =1,WETNESS IS CONSTANT; =2,FLUX=CONDUCTIVITY
THE FOLLOWING INPUT PARAMETERS ARE USED WHEN SOLVING THE HEAT BALANCE EQUATION (ITEMPS>0)
ALB  SURFACE SHORT WAVE ALBEDO 0.300000E 00
CLOUDS FRACTIONAL CLOUD COVER 0.0
CTRANS CLOUD TRANSMITTIVITY TO SHORT WAVES 0.200000E 00
SUNDEC SUN DECLINATION ANGLE (DEGREES) 0.0
XLAT  LATITUDE 0.335000E 02
TURB  TURBIDITY FACTOR 0.200000E 01
CEVAP0 COEFFICIENT FOR PET CALCULATION 0.0
CEVAP1 COEFFICIENT FOR PET CALCULATION 0.500000E-06
THESE ARE USED TO COMPUTE POTENTIAL EVAPOTRANSPIRATION AS A FUNCTION OF WIND SPEED (W, CM/SEC)
AND VAPOR PRESSURE DIFFERENCE (DVAP, MB.). THE MODEL IS PET = -(CEVAP0 + CEVAP1*W*DVAP)

```

Figure 4-4b.

TIME (DDHHMMSS) = 120000.00
 CURRENT INTEGRATOR STEP SIZE (SECONDS) = 0.2812500000 C2

FLUX COMPUTED AT	I-1/I BOUNDARY:	DPDZ COMPUTED AT I/I+1 BOUNDARY	DPDZ	DWOT	SINK	
I	DEPTH	WETNESS	FLUX	HYD COND	PHIAD	
1	0.250000	00	0.14449D 00	-0.52689D-05	0.50327D-10	-0.15721D 04
2	0.750000	00	0.19491D 00	-0.26452D-05	0.24646D-03	-0.37436D 03
3	0.200000	01	0.22710D 00	-0.55736D-06	0.17580D-07	-0.17432D 03
4	0.400000	01	0.24115D 00	-0.48969D-06	0.39238D-07	-0.12712D 03
5	0.750000	01	0.25532D 00	-0.36331D-06	0.82427D-07	-0.97054D 02
6	0.125000	02	0.26476D 00	-0.23461D-06	0.13211D-06	-0.80750D 02
7	0.175000	02	0.26787D 00	0.43155D-08	0.15380D-06	0.91925D 00
8	0.250000	02	0.27100D 00	0.63664D-07	0.17883D-06	0.57375D 00
9	0.350000	02	0.26613D 00	0.26735D-06	0.14134D-06	-0.68240D 00
10	0.450000	02	0.25865D 00	0.26221D-06	0.97563D-07	-0.12086D 01
11	0.550000	02	0.24836D 00	0.23497D-06	0.57557D-07	-0.20470D 01
12	0.650000	02	0.23638D 00	0.18002D-06	0.30256D-07	-0.31269D 01
13	0.750000	02	0.21449D 00	0.19159D-06	0.30256D-07	-0.89239D 01
14	0.850000	02	0.17759D 00	0.17275D-06	0.85574D-08	-0.36417D 02
15	0.950000	02	0.15300D 00	0.26971D-07	0.73531D-09	-0.59110D 03
				0.26971D-07	0.10591D-09	-0.12360D 04
						-0.65989D 02
						0.0

CUMULATIVE WETNESS VARIABLES (CM CF WATER)
 TOTAL IN PROFILE = 0.2330202460D 02

BOUNDARY	CUMULATIVE FLUX
1	-0.4414223714D-01
2	-0.3319122841D-01
16	0.8354746017D-03

EVAPOTRANSPIRATION OUTPUT - FLUXES IN CM/SEC
 TOTAL = -0.52689D-05 SOIL EVAP = -0.52689D-05 PLANT TRANSPIRATION = 0.0 CROWN POTENTIAL (CM) = 0.0

I	DEPTH	TEMP	FLUX	TCOND	TDIFF	WDIFF	VHCAP	UTDZ	OTEMPDT
1	0.250000	00	0.30449D 03	0.74955D-02	0.13864D-02	0.98134D-07	0.12742D-11	0.41958D 00	-0.74184D 01
2	0.750000	00	0.30078D 03	0.10480D-01	0.14396D-02	0.69057D-07	0.86253D-12	0.46998D 00	-0.44200D 01
3	0.200000	01	0.29525D 03	0.63892D-02	0.14693D-02	0.49797D-07	0.54546D-12	0.50217D 00	-0.28648D 01
4	0.400000	01	0.28952D 03	0.42215D-02	0.14778D-02	0.39761D-07	0.36080D-12	0.51621D 00	-0.14836D 01
5	0.750000	01	0.28433D 03	0.21989D-02	0.14930D-02	0.36378D-07	0.24242D-12	0.53038D 00	-0.93662D-01
6	0.125000	02	0.28386D 03	0.14060D-03	0.15093D-02	0.40310D-07	0.22383D-12	0.53981D 00	0.17114D 00
7	0.175000	02	0.28472D 03	-0.25890D-03	0.15162D-02	0.43058D-07	0.23250D-12	0.54292D 00	0.12193D 00
8	0.250000	02	0.28563D 03	-0.18516D-03	0.15233D-02	0.45782D-07	0.24224D-12	0.54605D 00	0.43554D-01
9	0.350000	02	0.28607D 03	-0.66170D-04	0.15152D-02	0.44232D-07	0.25580D-12	0.54119D 00	0.38028D-01
10	0.450000	02	0.28645D 03	-0.57379D-04	0.15025D-02	0.40332D-07	0.27269D-12	0.53371D 00	0.34513D-01
11	0.550000	02	0.28679D 03	-0.51552D-04	0.14849D-02	0.37176D-07	0.29369D-12	0.52342D 00	0.32606D-01
12	0.650000	02	0.28712D 03	-0.48083D-04	0.14645D-02	0.34717D-07	0.31760D-12	0.51144D 00	0.31988D-01
13	0.750000	02	0.28744D 03	-0.46251D-04	0.14274D-02	0.33240D-07	0.35723D-12	0.48956D 00	0.32533D-01
14	0.850000	02	0.28776D 03	-0.45432D-04	0.13661D-02	0.35213D-07	0.42161D-12	0.45267D 00	0.32546D-01
15	0.950000	02	0.28810D 03	-0.45148D-04	0.13267D-02	0.38702D-07	0.46921D-12	0.42809D 00	0.33546D-01
									0.0

TERMS OF THE HEAT BALANCE EQUATION: FLUXES IN CAL/CM**2/SEC
 NET RADIATION = 0.108938E-01 ET FLUX = -0.308758E-02 SENSIBLE HEAT = -0.310664E-03 HEAT ABSORBED BY SOIL = 0.749553E-02

ITERATIVE SOLUTION OF HEAT BALANCE EQUATION GIVES SURFACE TEMPERATURE = 0.305802D 03 DEGREES K

CUMULATIVE HEAT VARIABLES (CAL/CM**2)
 TOTAL IN PROFILE = 0.1457248556D 05

BOUNDARY	CUMULATIVE FLUX
1	0.5820627614D 02

Figure 4-5. Example of table output

the first integrated flux is the integrated soil evaporation rate. The last entry is the amount of water lost through the bottom boundary.

The evapotranspiration in cm/sec is shown in the next line. Values are given for the total flux (E , Equations 2-34 or 2-50), the soil evaporation flux (E_s , Equation 2-63 if a root model is included; otherwise it equals the total flux), and the plant transpiration (E_{p1} , Equation 2-64 if the root model is included). The crown potential (ϕ_p , Equation 2-61) is also given here.

The remaining output in this page gives values for parameters related to the temperature model. The first three columns are layer index, depth to center of the layer, and layer temperature ($^{\circ}\text{K}$). The next column is the heat flux (q , Equation 2-75b) at layer boundaries. The next four columns are: thermal conductivity (λ , Equation 2-23); thermal diffusivity (D_T , Equation 2-20c); vapor wetness diffusivity (coefficient of $\frac{d\psi}{d\theta}$ in the definition of $D_{\theta\text{vap}}$, Equation 2-15c); and the volumetric heat capacity (C , Equation 2-29). The derivative of temperature with respect to depth at the interior layers ($(\frac{dT}{dz})_i$, Equation 2-73b) is given in the next column. Finally, the derivative of temperature with respect to time ($(\frac{dT}{dt})_i$, Equation 2-72b), which is used to integrate the state equations, is given in the last column.

The values of the four terms of the heat balance equation (Equation 2-34) are given in the next line. These are net radiation (R , Equation 2-36), heat carried by the evapotranspiration flux (LE , Equation 2-42), sensible heat flux (H , Equation 2-46), and the heat absorbed by the soil (S , Equation 2-34). The surface temperature (T_s , Equation 2-47) is given in the next line.

Finally, cumulative heat fluxes can be optionally computed. NHCUMS is the number of such fluxes, and the array IXHCUM contains the boundary indices. In this example, only the integrated surface flux is computed.

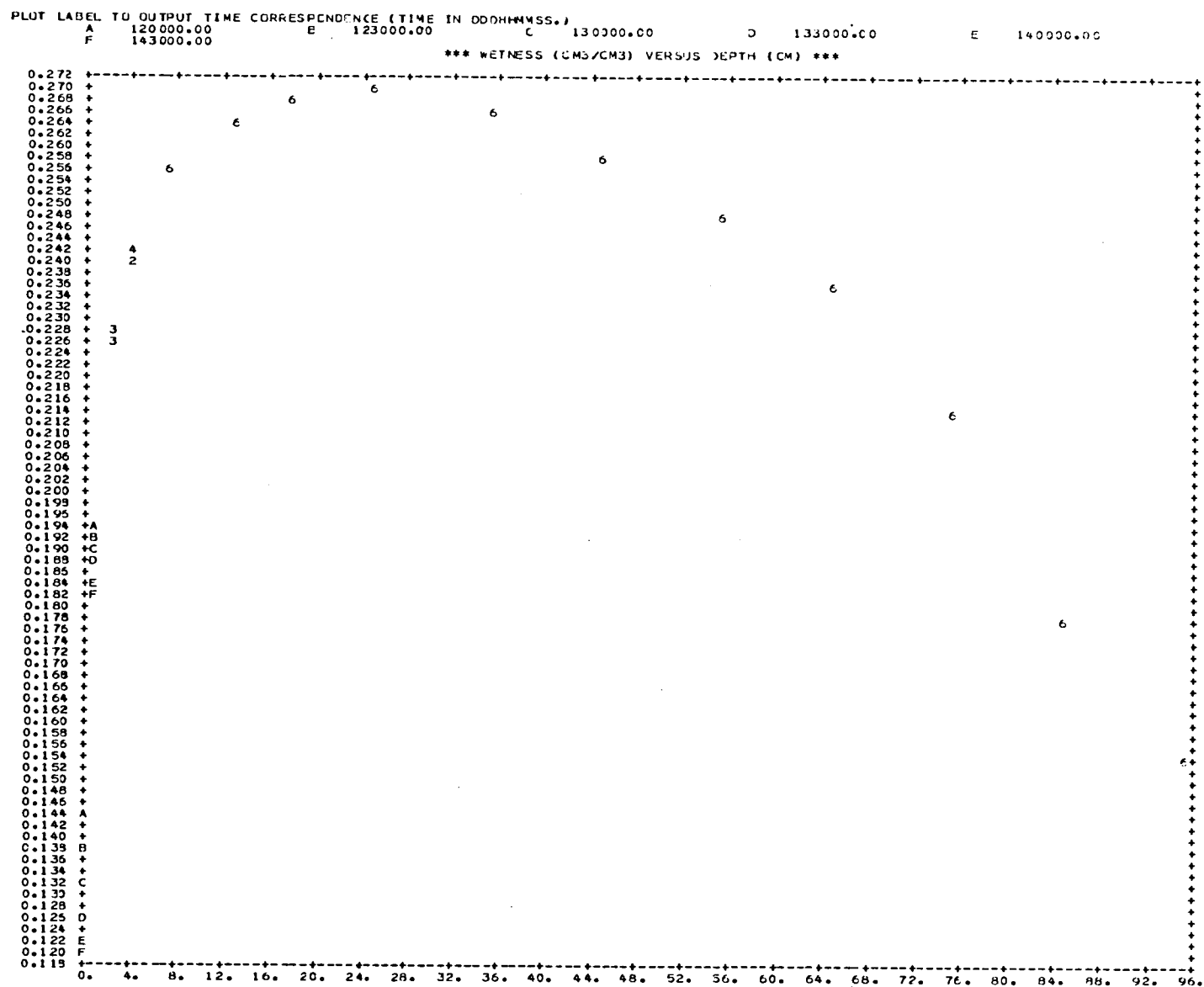
The next six figures are examples of the plots generated by the simulator. The FORTRAN unit number of the output is the value of the NAMELIST variable IPR.

Figures 4-6 and 4-7 are moisture and temperature profiles respectively. To reduce the amount of printout, more than one profile can be put on a plot page. Variable NFUNCT is the number of such graphs per plot page; in this example NFUNCT = 6. If NFUNCT is zero, no profiles will be plotted. The correspondence between print character and output time is shown at the top of each page. If more than one print character fall in the same place, then the number of such characters is plotted. These two figures show that over the time interval from noon to 2:30 PM, the profiles change only in the surface layers.

Figures 4-8 and 4-10 are plots of wetness and temperature respectively in selected layers as functions of time. NWATRS is the number of layers for which wetness is to be plotted, and INDXW is the array containing the indices of the chosen layers. In Figure 4-8 NWATRS is 3, and the indices are 1, 2, and 3. Similarly, NTEMPS is the number of layers for which temperature is to be plotted, and INDXT is the array containing the indices of the layers. In Figure 4-10 NTEMPS is 3, and the first three elements of INDXT are 1, 2, and 3.

Figures 4-9 and 4-11 are plots of moisture and heat fluxes at selected boundaries. NWFLUX is the number of boundaries for which wetness flux is to be plotted, and the array INDXWF contains the indices of those boundaries. NTFLEX and INDXTF are the number of boundaries and their indices for the heat flux plot.

The user can optionally control the vertical limits on all plots. NAMELIST variables WL and WH are respectively the lower and upper limits on wetness plots. (Figures 4-6 and 4-8). If WL = WH, then these values are ignored and appropriate limits are chosen from the data being plotted. Similarly, TL and TH are lower and upper limits for temperature plots (Figures 4-7 and 4-10); WFL and WFH are lower and upper limits for wetness flux plots (Figure 4-9); and TFL and TFH are lower and upper limits for heat flux plots. If any pair of lower and upper limits are equal, the actual limits are chosen from the data.



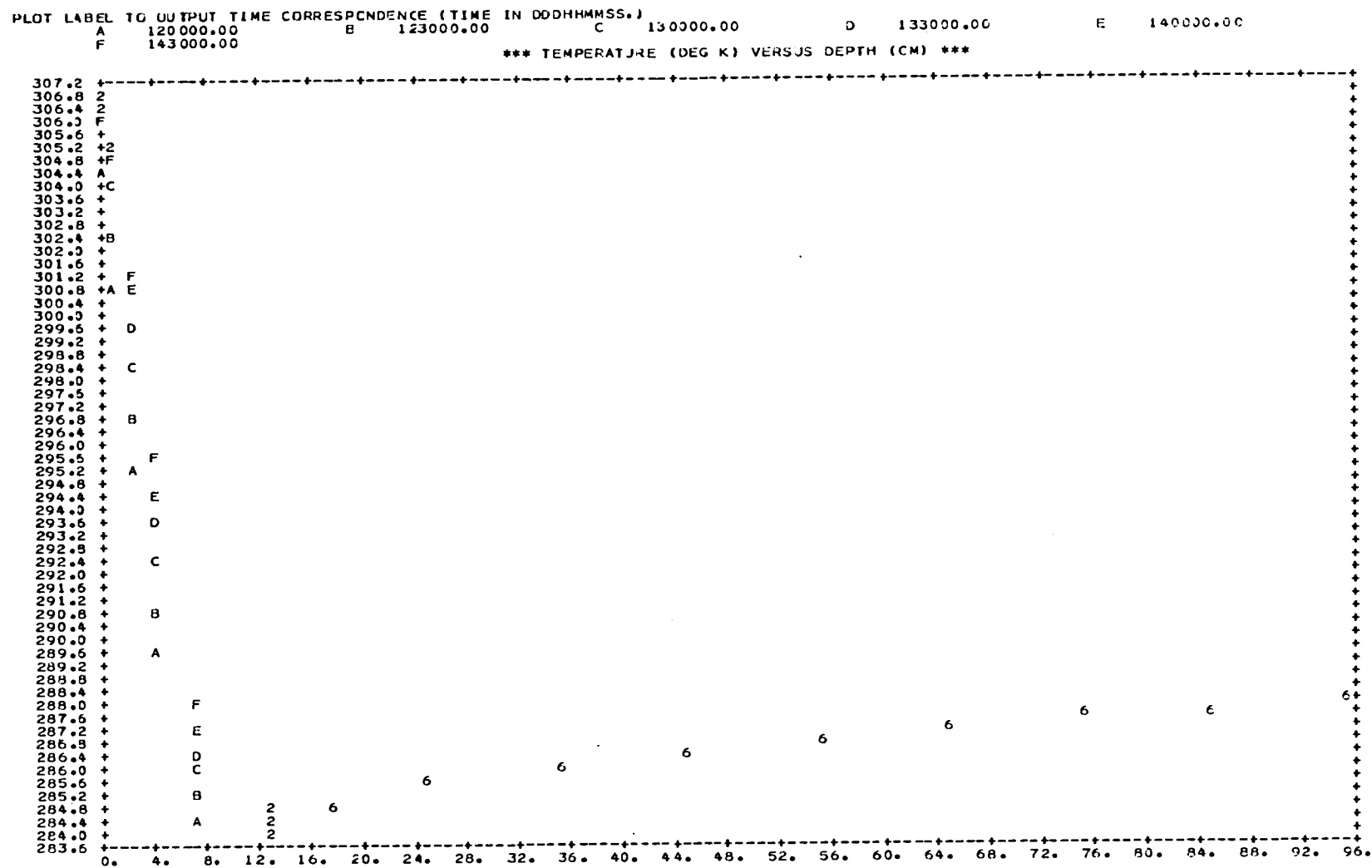


Figure 4-7. Example of plots of soil temperature profiles

Figure 4-8. Example of plots of soil moisture versus time in selected layers

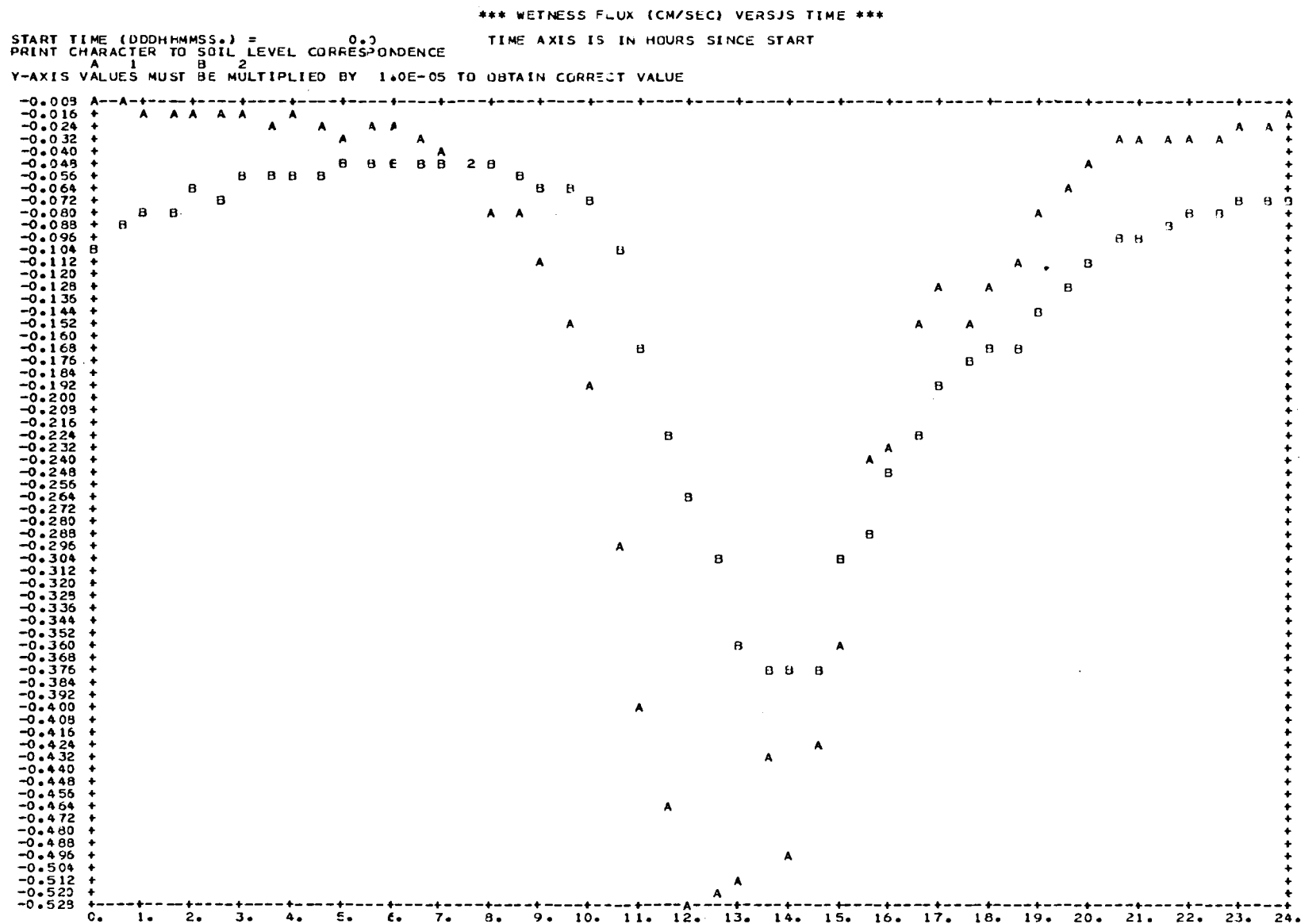


Figure 4-9. Example of plot of soil moisture flux versus time at selected boundaries

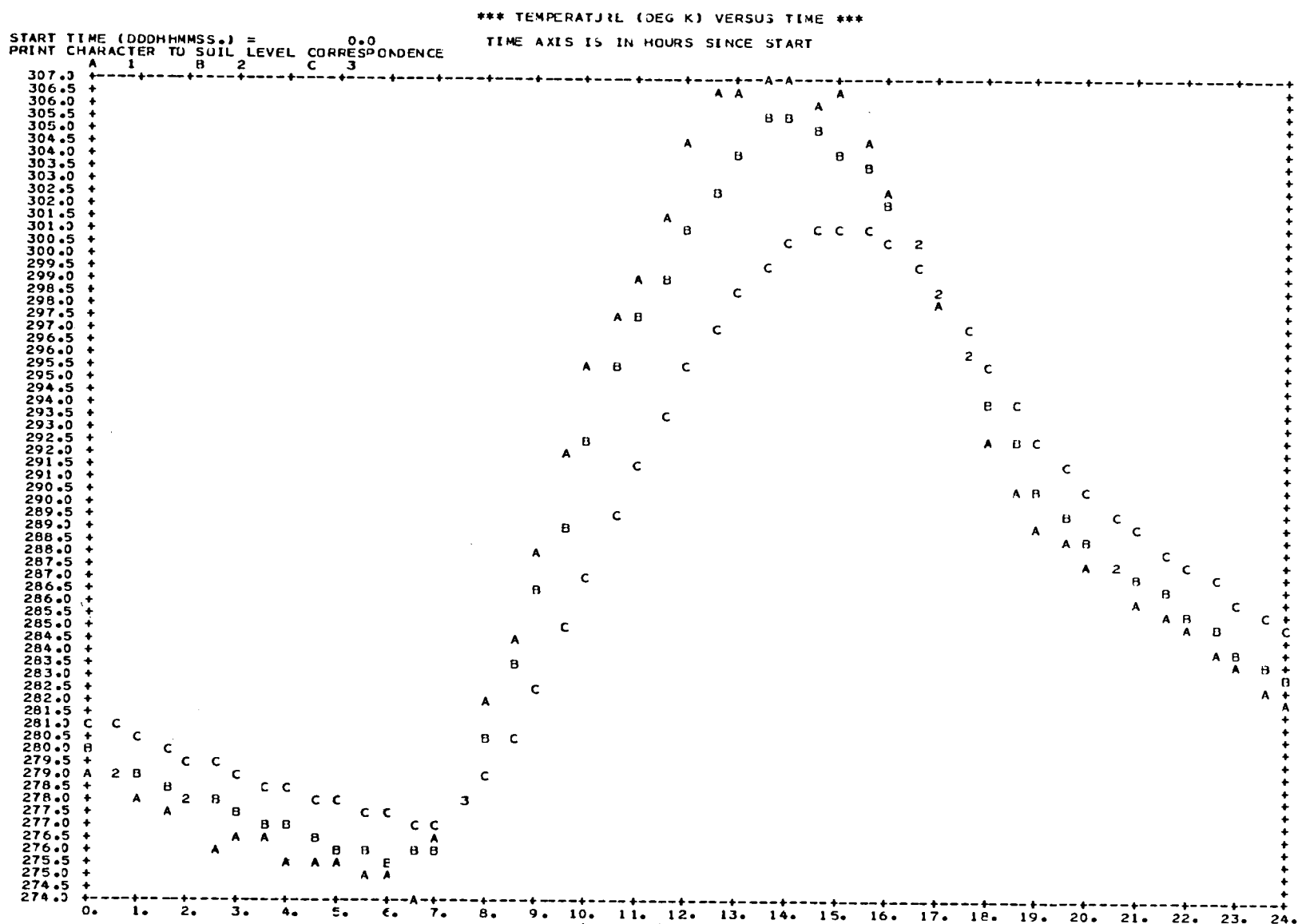


Figure 4-10. Example of plots of temperature versus time in selected layers

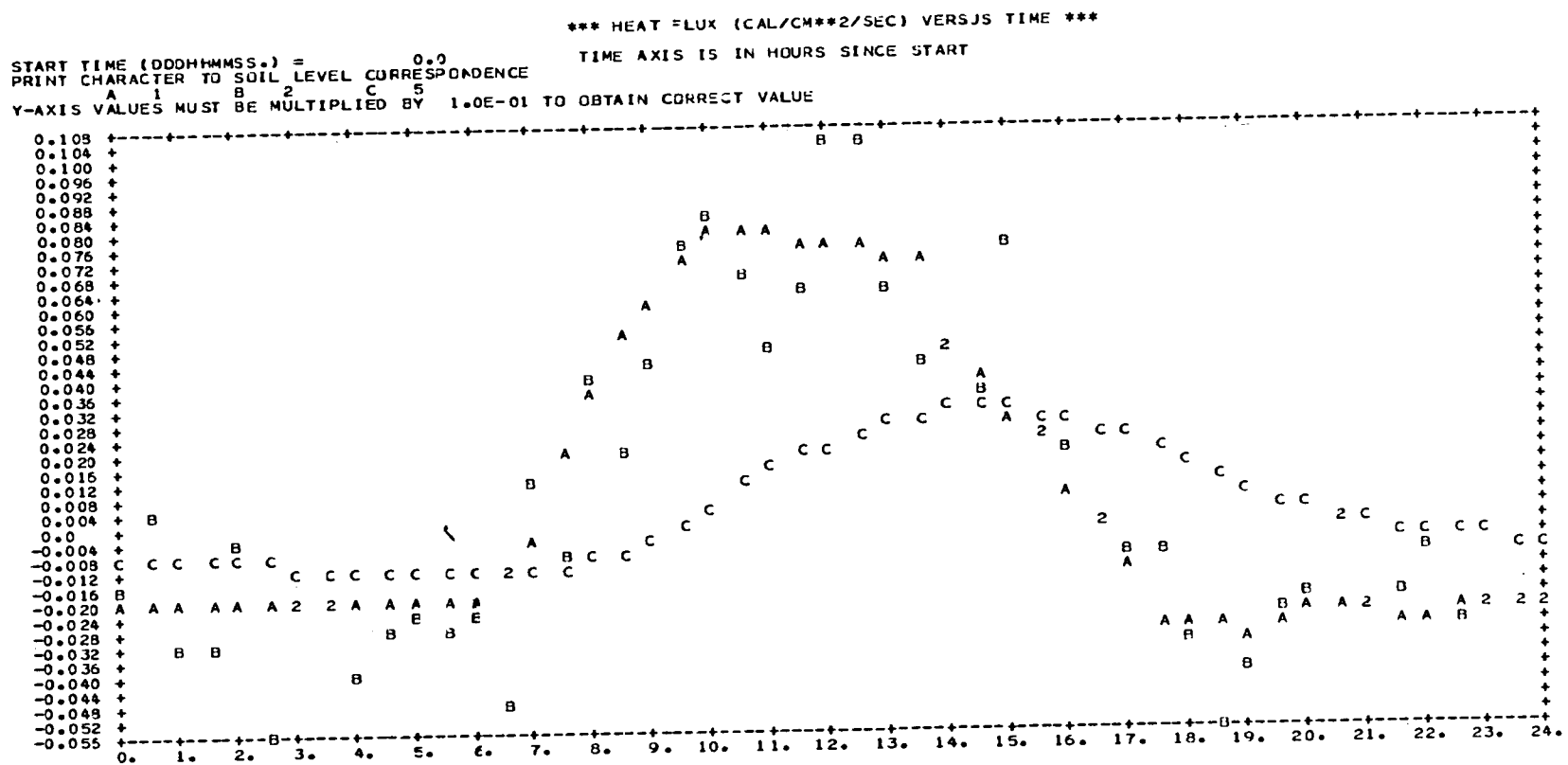


Figure 4-11. Example of plots of heat flux versus time at selected layer boundaries

Moisture and temperature profiles can also be output to disk or tape. The NAMELIST variable NDISK controls whether this is done, and IUDISK is the FORTRAN unit number of the DD card which points to the output data set. This can be a sequential data set on disk or tape, or a member of a partitioned disk data set.

The output records are unformatted. The record length is $8 * (NL + 1)$ bytes, where NL is the number of soil layers. This is $2 * NL + 2$ words of data per record. The records are written with subroutine FWRITE of the FTIO package. They can be read with subroutine FREAD from the same package. If desired, a FORTRAN read statement with a character format can be used instead.

One header record and one data record at each simulator output time are created. The first record contains the number of soil layers and thickness of each layer. Each data record contains the output time, the wetness profile, and the moisture profile. Record formats are shown in Figure 4-12.

Header Record	
Words	Contents
1	Binary zero
2	Number of soil layers, Integer*4
3	Layer thicknesses, Real*4
↓	↓
NL + 2	Zero (Real*4)
↓	↓
NL + 3	
↓	
2NL + 2	

Data Record	
Words	Contents
1-2	Output time (DDDHHMMSS.SS) Real*8
3	Wetness, Real*4
↓	↓
NL + 2	Temperature, Real*4
↓	↓
NL + 3	
↓	
2NL + 2	

Figure 4-12. Record format for profile data set created by the simulator

References

- Baginski, F., Galasso, R. E., McGarry, M., Nacios, G., and Kahn, J. CSC/SD-79/6080, "Solar Maximum Mission (SMM) Truth Model Attitude Simulator Algorithm Description and Operating Guide", Computer Sciences Corp., 1979.
- A. D. Booth, Numerical Methods. New York: Academic Press, 1957.
- Clapp, R. B. and Hornberger, G. M., "Empirical Equations for Some Soil Hydraulic Properties," Water Resources Res. 14: 601-604, 1978.
- de Vries, D. A., Heat Transfer in Soils in "Heat and Mass Transfer in the Biosphere." Washington, D. C.: Scripta Book Company, 1975.
- Eagleson, Peter S., Dynamic Hydrology, New York: McGraw Hill, 1970.
- Feddes, R. A., Bresler, E., and Neuman, S. P., "Field Test of a Modified Numerical Model for Water Uptake by Root Systems," Water Resources Res. 10: 1199-1206, 1974.
- Gardner, W. R., "Relation of Root Distribution to Water Uptake and Availability," Agronomy Journal 56: 41-45, 1964.
- Glang, R. D., "Partial Step Integration," Communications of the ACM, vol. 1, p. 521, 1966.
- Hillel, D., Soil and Water: Physical Principles and Processes. New York: Academic Press, 1971.
- Hillel, D., Computer Simulation of Soil Water Dynamics: A Compendium of Recent Work, International Development Research Centre, 1977.
- Jackson, R. D., Diurnal Changes in Soil-Water Content During Drying in "Field Soil Water Regime," R. R. Bruce et al. (eds.), Soil Science Society of America Proceedings, Special Publication No. 5, 1973.
- Jackson, Reginato, Kimball, and Nakayama, "Diurnal Soil-Water Evaporation: Comparison of Measured and Calculated Soil Water Fluxes," Soil Science Society of America Proceedings, vol. 38, pp. 861-866, 1974.
- Kimball, Jackson, Reginato, Nakayama, and Idso, "Comparison of Field-Measured and Calculated Soil-Heat Fluxes," Soil Sciences Society of America Journal, vol. 40, pp. 18-25, 1976a.
- Kimball, Jackson, Nakayama, Idso, and Reginato, "Soil-Heat Flux Determination: Temperature Gradient Method with Computed Thermal Conductivities," Soil Science Society of American Journal, vol. 40, pp. 25-28, 1976b.
- Lin, J. D., "On the Force-Restore Method for Prediction of Ground Surface Temperature," Journal of Geophysical Research, vol. 85, pp. 3251-3254, 1980.

Nielson, Jackson, Cary, and Evans (eds), Soil Water. Madison (Wisconsin): American Society of Agronomy and Soil Science Society of America, 1972.

Nimah, M. N. and Hanks, R. J., "Model for Estimating Soil Water, Plant, and Atmospheric Interrelations: I. Description and Sensitivity," Soil Sci. Soc. Am. Proc. 37: 522-527, 1973.

Philip, J. R., and de Vries, D. A., "Moisture Movement in Porous Materials Under Temperature Gradients," Transactions of the American Geophysical Union, vol. 38, pp. 222-228, 1957.

Philip, J. R., "General Method of Exact Solution of the Concentration Dependent Equation," Australian Journal of Physics, vol. 13, pp. 1-12, 1960.

Schmugge, T., "Remote Sensing of Surface Soil Moisture," Journal of Applied Meteorology, vol. 17, pp. 1549-1557, 1978.

Teddington, Modern Computing Methods. New York: Philosophical Library, Inc., 1958.

Whisler, F. D., Klute, A., and Millington, R. J., "Analysis of Steady State Evapotranspiration From a Soil Column," Soil Sci. Soc. Am. Proc. 32: 167-174, 1968.

BIBLIOGRAPHIC DATA SHEET

1. Report No. TM 82121	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle A Computer Program for the Simulation of Heat and Moisture Flow in Soils		5. Report Date April, 1981	
		6. Performing Organization Code 924	
7. Author(s) P. Camillo and T. J. Schmugge		8. Performing Organization Report No.	
9. Performing Organization Name and Address NASA Goddard Space Flight Center Greenbelt, MD 20771		10. Work Unit No.	
		11. Contract or Grant No. NAS 5-24350	
		13. Type of Report and Period Covered Technical Memorandum	
12. Sponsoring Agency Name and Address Hydrological Sciences Branch NASA/Goddard Space Flight Center Greenbelt, MD 20771		14. Sponsoring Agency Code	
15. Supplementary Notes This effort was supported by the Soil Moisture Project of the AgRISTARS program.			
16. Abstract This document describes a computer program that simulates the flow of heat and moisture in soils. The space-time dependence of temperature and moisture content is described by a set of diffusion-type partial differential equations. The simulator uses a predictor/corrector to numerically integrate them, giving wetness and temperature profiles as a function of time. The simulator was used to generate solutions to diffusion-type partial differential equations for which analytical solutions are known. These equations include both constant and variable diffusivities, and both flux and constant concentration boundary conditions. In all cases, the simulated and analytic solutions agreed to within the error bounds which were imposed on the integrator. Simulations of heat and moisture flow under actual field conditions were also performed. Ground truth data were used for the boundary conditions and soil transport properties. The qualitative agreement between simulated and measured profiles is an indication that the model equations are reasonably accurate representations of the physical processes involved.			
17. Key Words (Selected by Author(s)) Moisture & Heat Flow in Soils Computer Simulation Soil Moisture		18. Distribution Statement STAR Category 43 Unclassified-Unlimited	
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages	22. Price*

